

UNIVERSITÉ DU QUÉBEC À TROIS-RIVIÈRES

MÉMOIRE PRÉSENTÉ À
L'UNIVERSITÉ DU QUÉBEC À TROIS-RIVIÈRES

COMME EXIGENCE PARTIELLE
DE LA MAÎTRISE EN GÉNIE ÉLECTRIQUE

PAR

ORFAN ABOULSHAMAT

ALGORITHME DE COMMANDE ADAPTATIVE PAR MODE DE
GLISSEMENT À BASE DE RÉSEAUX DE NEURONES POUR UNE
ARTICULATION FLEXIBLE AVEC NON-LINÉARITÉ DURE

AVRIL 2003

Université du Québec à Trois-Rivières

Service de la bibliothèque

Avertissement

L'auteur de ce mémoire ou de cette thèse a autorisé l'Université du Québec à Trois-Rivières à diffuser, à des fins non lucratives, une copie de son mémoire ou de sa thèse.

Cette diffusion n'entraîne pas une renonciation de la part de l'auteur à ses droits de propriété intellectuelle, incluant le droit d'auteur, sur ce mémoire ou cette thèse. Notamment, la reproduction ou la publication de la totalité ou d'une partie importante de ce mémoire ou de cette thèse requiert son autorisation.

Dédicace

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

وَقُلْ رَبِّ زِدْنِي عِلْمًا

Et Dis : Mon Seigneur, fais moi croître en connaissance.
Qauran Sourat 20. Ayat 114

Tables des Matières

Résumé	v
Remerciements	vii
Liste des figures	viii
Liste des tableaux	xi
Liste des abréviations et des symboles	xii
1. Introduction	1
1.1 Problématique	2
1.1.1 Flexibilité structurelle dans les systèmes électromécaniques	2
1.1.2 Problème de commande	3
1.1.3 Approches de commande	4
1.1.4 Modélisation	6
1.2 Objectifs	8
1.3 Méthodologie	9
1.4 Organisation du mémoire	12

2. Modélisation	13
2.1 Modélisation d'une articulation flexible avec friction	14
2.1.1 Modèles utilisés	17
2.1.1.1 Modélisation du moteur et de la charge	17
2.1.1.2 Modélisation des phénomènes de friction	18
2.1.1.3 Modélisation complète de l'articulation	23
2.2 Commande inverse des systèmes	25
2.2.1 Modélisation inverse	25
2.2.2 Commande inverse	26
2.3 Conclusion	27
 3. Commande des systèmes électromécaniques	 28
3.1 Problématiques reliés à la commande des systèmes électromécaniques	29
3.1.1 Compensation de friction	29
3.1.2 Compensation de flexion	36
3.2 Méthodes modernes de commande pour les systèmes non-linéaires	37
3.2.1 Réseaux de neurones artificiels (RNA)	38
3.2.1.1 Principes de base des RNA	39

3.2.1.1.1 Neurone formel	39
3.2.1.2 Topologies de RNA	40
3.2.1.2.1 Les RNA récurrents	43
3.2.1.2.2 Les RNA multicouches	44
3.2.1.2.3 Les RNA Hopfield	45
3.2.1.3 RNA pour la commande	46
3.2.2 Commande par mode de glissement	47
3.2.3 Logique floue	52
3.3 Conclusion	60
 4. Commande de position d'une articulation flexible avec friction	 61
4.1 Commande inverse des articulations flexibles avec friction	63
4.1.1 Modélisation inverse avec RNA	63
4.2 Algorithme de commande inverse	68
4.2.1 Développement théorique	68
4.3 Conclusion	78

5. Résultats de simulation et analyse de performances	79
5.1 Performances dynamiques et statiques de la loi de commande	80
5.1.1 Réponse à la consigne sans le contrôleur	81
5.1.2 Réponse à la consigne avec contrôleur MG	84
5.1.3 Réponse à la consigne avec contrôleur MG avec une rétroaction du terme de vitesse du moteur	87
5.2 Conclusion	93
 6. Conclusion	 94
 Bibliographie	 97
 Annexe A : Programmation du Réseau de neurones	 100
 Annexe B : Approximation de la fonction "sign"	 102
 Annexe C : Syntonisation des gains du contrôleur MG	 104

Résumé

Dans ce mémoire nous présentons le développement d'un algorithme de commande d'une articulation flexible qui est un système électromécanique sous-actionné. La performance des systèmes d'entraînement est limitée par les propriétés des éléments de transmission mécanique (courroies, réducteurs de vitesse, etc.). La flexibilité, le frottement sec et le retour des dents sont des effets qui peuvent causer des vibrations indésirables qui réduisent la vie utile des équipements et causent une dégradation des performances des manipulateurs de grande précision, opérant à de grandes vitesses ou sous des conditions de charges lourdes.

L'objectif général du projet de recherche concerne le développement d'une méthodologie de commande adaptative de systèmes dynamiques non linéaires à l'aide de réseaux de neurones artificiels, en particulier les articulations flexibles, afin d'améliorer les performances dynamiques. La propriété d'apprentissage des réseaux de neurones est exploitée pour dériver le modèle inverse des articulations flexibles.

Pour établir et valider la méthodologie, une articulation flexible, avec caractéristique de flexion linéaire et avec non-linéarités dures, causées par la présence de friction, est d'abord modélisée. La conception du contrôleur développé est composée de deux étapes:

1- Le développement d'un modèle inverse du système en utilisant un réseau de neurone artificiel (RNA) multicouches qui va compenser pour la non-linéarité et générer le signal d'anticipation.

2- Le développement d'un stabilisateur rétroactif par mode de glissement, avec une bande d'adoucissement adaptative implantée par la logique floue.

Les résultats sont validés par simulation. La robustesse du contrôleur, contre la variation des paramètres du système et le changement des conditions initiales, est démontrée.

Remerciements

Le travail présenté dans ce mémoire a été réalisé au laboratoire de commande de l'UQTR sous la direction de monsieur Pierre Sicard. Je tiens à le remercier sincèrement pour son soutien et sa confiance. Son jugement et ses connaissances m'ont permis d'acquérir des qualités inestimables en recherche, soit la rigueur scientifique, la ténacité, la perfection et le sens critique.

Je tiens à remercier tous mes professeurs qui m'ont donné sans compter leurs connaissances tout au long de mes longues études.

Toute ma reconnaissance à mes parents qui m'ont toujours soutenu, guidé et encouragé. Mes frères et mes amis et collègues avec qui j'ai apprécié la joie de savoir et de se construire.

À Ghalia source de joie, de calme et surtout d'amour. Elle a été d'une compréhension exemplaire. Elle a cru en moi et en nous, son support et ses conseils ont toujours été d'un grand secours.

Liste des figures

Figure 2.1 : Structure de l'articulation flexible	16
Figure 2.2 : Modèle du moteur et de la charge	17
Figure 2.3 : Modèles simples de friction	19
Figure 2.4 : Modèle de friction retenu	21
Figure 2.5 : Algorithme de calcul du couple de frottement	22
Figure 2.6 : Modélisation de l'articulation flexible	24
Figure 2.7 : Modèles direct et inverse	25
Figure 2.8 : Commande inverse	27
Figure 3.1 : Système comprenant le module de friction	31
Figure 3.2 : Consignes de position	31
Figure 3.3 : Effets de la friction sur la réponse indicielle	32
Figure 3.4 : Effets de la friction sur la réponse sinusoïdale	33
Figure 3.5 : Neurone formel	40
Figure 3.6 : Schématisation d'un RNA multicouche	40
Figure 3.7 : Fonction d'activation du neurone à seuil (a) et du neurone linéaire (b)	42
Figure 3.8 : Variables linéairement séparables (a) et linéairement non séparables	42
Figure 3.9 : Fonctions d'activation, a) Sigmoïde et b) Fonction « Linéaire par partie »	43
Figure 3.10 : Schématisation d'un RNA récuratif (NARX)	44
Figure 3.11 : Réseau de neurones multicouche	45
Figure 3.12 : Réseau Hopfield	46
Figure 3.13 : Représentation de la logique classique	53
Figure 3.14 : Représentation de la logique floue	54

Figure 3.15 : Fonctions d'appartenance en triangle	54
Figure 3.16 : Fonctions d'appartenance en cloche	55
Figure 3.17 : Fonctions d'appartenance en trapèze	55
Figure 3.18 : Connecteurs de précision	56
Figure 3.19 : Résultat d'application des règles floues	57
Figure 3.20 : Technique du centre de gravité	58
Figure 4.1 : Architecture d'apprentissage général	64
Figure 4.2 : Trajectoire d'apprentissage	65
Figure 4.3 : Fonction linéaire saturée	65
Figure 4.4 : Fonction Sigmoidale	66
Figure 4.5 : Fonction purelin	66
Figure 4.6 : Diagramme bloc du système	69
Figure 4.7 : Modèle Simulink® du système	69
Figure 4.8 : Effet de la variation de la constante de temps spécifiée sur le temps de convergence vers la droite de glissement suite à un échelon d'erreur	71
Figure 4.9 : Domaine des pôles imposés	71
Figure 4.10 : Définition du plan de phase pour un système avec caractéristique spécifiée du premier ordre	73
Figure 4.11 : Structure de contrôleur en MG	77
Figure 5.1 : Position et vitesse de consigne	80
Figure 5.2 : Résultats avec RNA d'anticipation	82
Figure 5.3 : La vitesse du moteur avec RNA d'anticipation	82
Figure 5.4 : Couple d'anticipation à la sortie du RNA	83

Figure 5.5 : Erreur de position avec RNA d'anticipation_____	83
Figure 5.6 : Résultats avec RNA d'anticipation_____	84
Figure 5.7 : Couple de commande à la sortie du contrôleur MG_____	85
Figure 5.8 : Erreur de position avec contrôleur MG_____	85
Figure 5.9 : Résultats avec RNA et contrôleur MG et conditions initiales non nulles_	86
Figure 5.10 : Couple de commande à la sortie du contrôleur MG et conditions initiales non nulles_____	86
Figure 5.11 : Erreur de position avec contrôleur MG et conditions initiales non nulles_____	87
Figure 5.12 : Résultats avec RNA et contrôleur MG incluant un terme en vitesse du moteur_____	88
Figure 5.13 : Couple de commande à la sortie du contrôleur MG incluant la rétroaction de vitesse du moteur_____	88
Figure 5.14 : Erreur de position avec contrôleur MG incluant la rétroaction de vitesse du moteur_____	89
Figure 5.15 : Résultats avec RNA et contrôleur MG incluant un terme en vitesse du moteur et conditions initiales non nulles_____	89
Figure 5.16 : Couple de commande à la sortie du contrôleur MG incluant un terme en vitesse du moteur et conditions initiales non nulles_____	90
Figure 5.17 : Erreur de position avec contrôleur MG incluant un terme en vitesse du moteur et conditions initiales non nulles_____	90
Figure 5.18 : Robustesse du contrôleur face aux variations des paramètres et de la charge _____	92
Figure A.1 : Approximation floue de la fonction Signe sur Matlab®_____	102

Liste des tableaux

Tableau 5.1 Indices de performance	91
------------------------------------	----

Liste des abréviations et des symboles

Symboles

$\tilde{\bullet}$	Signal bruité
$\hat{\bullet}$	Signal estimé
X	Vecteur d'état $[x_1, x_2, x_3, \dots, x_n]$

Abréviations

ASIC	Circuit intégré à application spécifique (Application Specific Integrated Circuit)
CAO	Conception Assistée par Ordinateur
ITGE	Intégration à Très Grande Échelle
LMS	Moindres carrés moyens (Least Mean Squares)
MG	Mode de Glissement
RJF	Robots à Joints Flexibles
RLS	Moindres carrés récurrents (Recursive Least Squares)
RNA	Réseau de neurones artificiels
RSV	Régulateur à Structure Variable
SSV	Système à Structure Variable

Chapitre 1

Introduction

1.1 Problématique.

Les systèmes d'entraînement mécaniques de haute performance prennent de plus en plus d'importance dans les domaines d'automatisation et de production. La grande rapidité et la grande précision de suivi des trajectoires sont des capacités indispensables dans diverses applications.

1.1.1 Flexibilité structurelle dans les systèmes électromécaniques

Les performances de ces systèmes sont limitées par les propriétés des éléments mécaniques qui les constituent. Les performances dynamiques des systèmes sont fortement affectées par la flexibilité structurelle et par les non-linéarités dures des phénomènes de frottement entre autres [1].

Les systèmes d'entraînement qui emploient des boîtes d'engrenage ou un élément de transmission de couple quelconque, les systèmes d'entraînement hydrauliques et les manipulateurs de grande dimension ont une flexibilité inhérente au système d'entraînement ou à la structure qui est un facteur limitatif important des performances du système [1]. Ces systèmes possèdent plus d'axes de liberté qu'ils ont d'entrées de commande et font donc partie de la famille des systèmes sous-actionnés. Ceci implique que, entre autres, les lois de commande pour les manipulateurs dits rigides ne s'appliquent pas directement aux manipulateurs flexibles [2]. D'autre part la flexibilité ajoute des modes qui peuvent être excités par des forces externes.

Les manipulateurs de type bras de robots avec joints flexibles (RJF) sont soumis aux contraintes suivantes :

- La flexibilité due aux éléments de transmission et la dynamique non modélisée associée aux modes à hautes fréquences;
- Les phénomènes de frottement;
- Les incertitudes de structure sur certains paramètres (F_c, F_v, F_s, v_s) , où F_c, F_v, F_s sont respectivement les coefficients de frottements de Coulomb, visqueux et statique, v_s est la vitesse caractéristique de frottement statique;
- Variations temporelles de leurs paramètres;
- Des limites sur les valeurs des variables articulaires et sur les couples de commande.

L'écriture d'une loi de commande efficace pour de tels systèmes doit être réalisée en fonction de toutes ces contraintes.

1.1.2 Problème de commande

Lors de la conception d'un système de commande, les objectifs principalement considérés sont la précision, la rapidité de réponse, la stabilité et les caractéristiques en régulation ou en poursuite ou un critère de performance incorporant plus ou moins directement l'ensemble de ces objectifs et permettant d'obtenir une commande dite optimale. Un autre objectif très important est la robustesse ou l'insensibilité face aux variations de paramètres et aux perturbations.

Dans l'exemple d'une articulation de robot, l'importance de la robustesse est bien démontrée. Il est désirable que la réponse du système soit insensible aux variations de l'inertie qui dépend de la longueur du bras du robot, de la charge qu'il transporte et de la position spatio-temporelle du bras. Une sensibilité à ces variations peut causer l'instabilité de la commande, ce qui est particulièrement dommageable à haute vitesse.

La robustesse est aussi un critère important avec les machines outils (perturbations de couple et charge avec inertie variable) et dans plusieurs applications d'entraînements à vitesse variable : commande de vitesse des ascenseurs, entraînements en aciérie et entraînements multi machines dans l'industrie du papier et du textile [3].

1.1.3 Approches de commande

La commande de ces systèmes requiert la compensation de l'effet des non linéarités, l'adaptation face aux variations de paramètres et le rejet des perturbations.

Pour réaliser ceci, différentes fonctions peuvent être utilisées dont:

- (a) Rétroaction et anticipation (*feed-forward*)
- (b) Rétroaction non linéaire
- (c) Adaptation active
- (d) Adaptation passive

Le terme adaptatif peut être interprété de différentes façons mais implique habituellement que le système peut s'accommoder aux changements imprévisibles de l'environnement,

que ces changements proviennent du système ou lui soient externes. La commande adaptative (active) est définie comme étant un algorithme de commande comportant un certain mécanisme d'identification des paramètres pour l'adaptation. Sinon, on se trouve en présence d'une adaptation passive.

L'adaptation active est la solution classique à la commande des systèmes variant dans le temps/. Cependant, l'adaptation passive, en particulier la commande par correcteur à structure variable, est souvent considérée pour ce genre de problème [3]. Analysons les caractéristiques des deux types de commande adaptative (active et passive).

Il existe trois stratégies principales de commande adaptative active : programmation automatique des gains (*gain scheduling*), commande adaptative avec modèle de référence (*model reference adaptive control*) et les régulateurs auto syntonisant (*Self-tuning regulators*) [3].

D'excellents résultats peuvent être obtenus avec la commande adaptative avec modèle de référence et les régulateurs auto syntonisant auxquels seraient ajoutés la fonction anticipation pour rejeter les perturbations. Cependant, ces deux méthodes impliquent une quantité de calculs importante et ne sont donc pas appropriées à la commande des systèmes rapides. D'autre part, la programmation automatique des gains peut aussi donner de bonnes performances mais la phase de conception est très longue et ce type de commande n'est pas robuste.

Le correcteur adaptatif passif (selon la définition dans [3]) le plus répandu et le plus général est le régulateur à structure variable (RSV) opérant en mode de glissement (MG) (*sliding mode*). Ce type de système a été étudié d'abord en Union Soviétique [3]. Par la suite, ces travaux ont été repris ailleurs, soit pour compléter l'étude théorique, soit pour étudier quelques applications possibles [4].

La commande par correcteur fonctionnant en MG présente des discontinuités sur les hyperplans de glissement (plans de commutation) qui causent des oscillations (*chattering*) à la sortie du système et une activité intense du système de commande. Ces deux phénomènes sont amplifiés par la présence de faibles délais (fréquence de commutation limitée) et de faibles constantes de temps négligées lors de la conception [4]. Ainsi, alors que la commande en MG résulte en une loi de commande robuste face aux variations de paramètres et aux perturbations, elle n'est pas robuste face aux approximations usuelles de modélisation (les modes négligés qui résident hors de la plage de fréquence d'intérêt). Ainsi, l'effet de la commutation constante de la commande sur les éléments mécaniques du système doit être considéré [5]. De plus, l'instabilité du signal de commande peut causer une erreur stationnaire à la sortie. Ainsi, le signal de commande doit être traité afin d'atténuer ou d'éliminer les oscillations.

1.1.4 Modélisation

Le calcul du modèle dynamique et du modèle dynamique inverse du système d'entraînement mécanique de haute performance est très important pour la commande, la

simulation et la programmation du système. Le premier est pour dériver l'équation dynamique de mouvement décrivant le comportement dynamique du système mécanique. Le second est pour calculer la valeur de l'entrée de commande par anticipation, pour produire la trajectoire désirée du système.

L'articulation robotique est un exemple d'un système d'entraînement mécanique. Plusieurs méthodes pour dériver le modèle dynamique du manipulateur sont utilisées : Newton Euler, Lagrange-Euler, etc. [6] Ces méthodes sont convenables pour les robots à joints assez rigides qui demandent, pour donner la rigidité et éliminer les vibrations structurelles, l'utilisation de métal épais et fort en fabriquant les manipulateurs ce qui implique une masse plus élevée, plus de coût et plus d'énergie consommée pour accélérer le mécanisme du manipulateur. Par contre, les robots à joints flexibles ont moins de masse, consomment moins d'énergie et ils ont éventuellement plus de manœuvrabilité d'où l'intérêt d'étudier et développer des techniques de commande pour ces systèmes.

La modélisation dynamique de l'articulation flexible, incluant les calculs de dynamique inverse, est la première étape dans la conception d'un algorithme de commande [6]. Quand le modèle dynamique du système étudié n'est pas connu *a priori* (ou ce modèle n'est pas disponible) le système doit-être identifié. Après, une loi de commande est construite en se basant sur le modèle estimé. C'est l'idée de base derrière les stratégies de commande adaptative et par apprentissage. Pendant les dernières années, il y a eu un progrès considérable dans la compréhension des fonctionnalités qu'on peut atteindre avec

les réseaux de neurones artificiels, ainsi que les procédures et les architectures disponibles pour réaliser ces fonctionnalités.

1.2 Objectifs

Nous désirons ici déterminer une stratégie de commande bien adaptée à la commande des systèmes électromécaniques variant dans le temps et soumis à des perturbations, spécialement une articulation flexible. En particulier nous voulons développer une stratégie de commande de positionnement et de suivi précis des trajectoires d'un système électromécanique (c.-à-d. une seule articulation flexible avec frottement). Le système conçu devra être robuste face aux variations de paramètres et de la charge. Il devra aussi être rapide et offrir de bonnes performances statiques.

Pour ça nous désirons évaluer une stratégie basée sur l'utilisation des réseaux de neurones artificiels pour obtenir le modèle inverse du système qui va générer le signal d'anticipation (*feed-forward*) et un correcteur par mode de glissement qui permet d'obtenir une adaptation passive.

Le problème d'élimination des oscillations associées au réglage par mode de glissement sera particulièrement considéré. Nous voulons aussi évaluer une solution d'implantation avec la logique floue d'une approximation de la fonction discontinue « signe » pour éliminer ces oscillations.

Cet objectif est encadré dans les travaux de recherche effectués au groupe de recherche en électronique industrielle à l'UQTR dont l'objectif principal est de développer un algorithme de commande adaptative général pour les robots avec joints flexibles (RJF).

La théorie de la commande des systèmes non-linéaires et la connaissance des propriétés des RJF seront exploitées pour développer les algorithmes d'apprentissage. Ma contribution est dans l'application des RNA au problème de commande de RJF et par la compensation de non-linéarités dures dans un système sous-actionné.

1.3 Méthodologie.

D'abord, le comportement dynamique de l'articulation flexible doit être complètement modélisé ainsi que les phénomènes de frottements auxquels elle est soumise.

À partir de cette modélisation, la synthèse d'un contrôleur permettra d'assurer le suivi de trajectoire de l'articulation. Considérant les fortes non-linéarités des modèles de frottement, l'utilisation de contrôleurs linéaires classiques tels que les PID ou PD s'avère insuffisante. Il faudra donc concevoir une loi de commande qui contiendra un algorithme qui tient compte des paramètres de l'articulation. Les paramètres ainsi estimés permettront d'inclure dans la loi de commande une compensation efficace pour résoudre les problèmes cités.

Plusieurs lois de commande ont été proposées pour la commande de manipulateurs avec joints flexibles dont les techniques de perturbations singulières [7], la linéarisation par rétroaction [2], la commande robuste [7] [8] et la commande adaptative [9]. Ces travaux

considèrent un modèle qui exclut toutes les non-linéarités dures (frottement sec, retour de dents, etc.), tel que ces résultats demeurent souvent académiques. La complexité des modèles, la dynamique rapide associée aux non-linéarités et les incertitudes ou la variation des paramètres de l'articulation font des réseaux de neurones artificiels (RNA) d'excellents candidats pour la commande de ces systèmes.

La référence [10] sur l'identification de la dynamique inverse des systèmes non linéaires inconnus présente une classe d'algorithmes d'apprentissage pour les poids d'un réseau de neurones multicouches. On va se baser sur cette étude pour développer le modèle inverse de RJF avec friction et pour la génération d'un signal de commande d'anticipation. L'apprentissage prendra, dans notre travail, la forme hors ligne (l'apprentissage en ligne sera le sujet de recherches futures) avec un algorithme dérivé de l'approche de rétro-propagation.

À cause de la non-linéarité et de la discontinuité inhérente dans le modèle de l'articulation flexible avec friction, on ne peut pas rencontrer la condition d'unicité pour dériver le modèle inverse en utilisant les techniques conventionnelles. Pour contourner ce problème, un RNA apprendra le modèle inverse approximatif de l'articulation. Pour augmenter la performance de suivi des trajectoires de l'articulation flexible en minimisant l'erreur de position, un correcteur robuste en mode de glissement (MG), qui considère les effets de couplage entre les articulations comme des perturbations, permettra de concevoir un correcteur simple qui compense pour les imprécisions du modèle. Les correcteurs robustes par MG simplifient la programmation de haut niveau.

Malgré ces avantages incontestables, cette méthode a des désavantages comme la forte oscillation de l'organe de commande et la nécessité d'utiliser la grandeur à régler et un certain nombre de ses dérivées, selon l'ordre du système. De plus, il y a une phase de convergence pendant laquelle le système est sensible aux variations de paramètres et aux perturbations. Cependant, de nombreux travaux ont permis de développer différentes stratégies pour corriger ces faiblesses [3] :

- (a) ajout d'un filtre passe-bas à l'entrée du procédé;
- (b) régulation de la dérivée de l'erreur par rapport à l'hyperplan de glissement;
- (c) substitution du terme discontinu de la loi de commande par une fonction d'adoucissement. Cette fonction d'adoucissement est définie par :
 - (i) Une approximation continue du terme discontinu;
 - (ii) Une bande de largeur constante avoisinant l'hyperplan de glissement à l'intérieur de laquelle le gain est variable;
 - (iii) Une bande de largeur variable avoisinant l'hyperplan de glissement à l'intérieur de laquelle le gain est variable;

Nous proposons de réaliser la fonction d'adoucissement qui implique l'implantation d'une approximation de la fonction « signe » par la logique floue qui permet d'approximer la fonction « Signe » naturellement par une fonction continue dont on peut modifier la caractéristique par des règles empiriques.

Des simulations comportementales du système complet pour une trajectoire de consigne donnée permettront d'évaluer les performances de la loi de commande, en particulier la qualité de suivi de trajectoire et de compensation de friction.

1.4 Organisation du mémoire.

Le chapitre 2 présente le modèle d'un joint flexible avec friction. Ce modèle sera utilisé pour mettre le point sur les problématiques associées à la commande de la position des articulations flexibles. Les termes « modélisation inverse » ainsi que « commande inverse » sont définis.

Au troisième chapitre une description de la commande des systèmes électromécaniques non-linéaires, les genres de problèmes associés ainsi que les méthodes conventionnelles utilisées pour les résoudre (spécialement la compensation de friction et de flexion) et la place de l'intelligence artificielle dans le domaine de commande sont présentés. Le principe des réseaux de neurones artificiels est présenté en détail ainsi que leurs applications pour commander les systèmes non-linéaires.

Le quatrième chapitre présente l'algorithme de commande développé pour le positionnement des articulations flexibles. Les composantes de l'algorithme sont introduites, soit le modèle inverse neuronal et le contrôleur par mode de glissement avec une bande d'adoucissement adaptative implantée avec la logique floue, pour arriver à sa structure finale.

Au chapitre 5, les résultats de simulation sont présentés et la performance est analysée.

Des conclusions sont ensuite tirées au chapitre 6.

Chapitre 2

Modélisation

2.1 Modélisation d'une articulation flexible avec friction

Les robots considérés dans les études des systèmes de commande sont généralement constitués de plusieurs membres reliés par des articulations et mus par des actionneurs (moteurs), le plus souvent situés dans ces articulations. L'objectif général du contrôle d'un tel type de manipulateur est de lui faire réaliser des tâches précises. Il s'agit donc d'assurer le contrôle de la trajectoire du manipulateur dans son espace de travail. Cela nécessite un bon contrôle de chacune des articulations du robot afin que l'effecteur suive la trajectoire désirée. Plus précisément, les forces ou couples adéquats doivent être appliqués à chaque actionneur pour que l'effecteur suive la trajectoire désirée définie d'après la tâche à accomplir. Dans certains cas, il peut aussi être nécessaire de contrôler les vitesses et les accélérations de l'effecteur pour suivre également les vitesses et accélérations désirées [11]. Une articulation est le lien entre deux membres du manipulateur ou bien pour les extrémités, entre un membre et la base du robot ou le monde extérieur. Dans ce travail nous considérons une de ces articulations. L'actionneur est supposé être commandé en couple et le lien mu par l'actionneur est considéré comme une charge pour celui-ci. La stratégie de contrôle applicable à une articulation flexible est très dépendante de la structure du modèle retenu. Cette structure peut dépendre des caractéristiques particulières du système considéré ou de conditions particulières d'utilisation de celui-ci. En conséquence, les chercheurs ont très souvent développé leur propre modèle, satisfaisant leurs conditions et décrivant avec le plus d'exactitude possible le comportement de leur système [12].

Les équations dynamiques du mouvement d'un manipulateur à n articulations peuvent ainsi s'écrire sous la forme matricielle :

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + F(\dot{q}) + g(q) = \tau - J^T(q)h \quad (2.1)$$

Avec M la matrice des masses et inerties, C est la matrice des termes centrifuges et de Coriolis, F est la matrice des forces de frottements, g représente l'action de la gravité, τ la matrice des couples appliqués par les actionneurs, le terme $J^T(q)h$ représente l'action de l'effecteur sur son environnement et q désigne le vecteur des variables articulaires.

Les modèles d'articulations peuvent être déduits de cette équation en considérant les termes appropriés et en adaptant l'écriture des matrices au système étudié.

Dans cette étude, nous considérons une seule articulation flexible, qui consiste en un moteur commandé en couple pilotant une charge par le biais d'une boîte de transmission équivalente à un ressort de torsion [13]. Dans la suite, les indices M , L , et T désigneront respectivement les valeurs reliées au moteur, à la charge et à la transmission. L'indice F désigne les termes relatifs à la friction.

La figure 2.1 donne une représentation du modèle de l'articulation comprenant le moteur, la charge et la boîte de transmission.

Les équations mathématiques du modèle sont :

$$\begin{cases} \frac{dw_M}{dt} = J_M^{-1}(\tau_M - \tau_{FM} - \eta\tau_T) \\ \frac{d\theta_M}{dt} = w_M \\ \frac{dw_L}{dt} = J_L^{-1}(N\tau_T - \tau_{FL}) \\ \frac{d\theta_L}{dt} = w_L \\ \tau_T = k(\theta_M - N\theta_L) \end{cases} \quad (2.2)$$

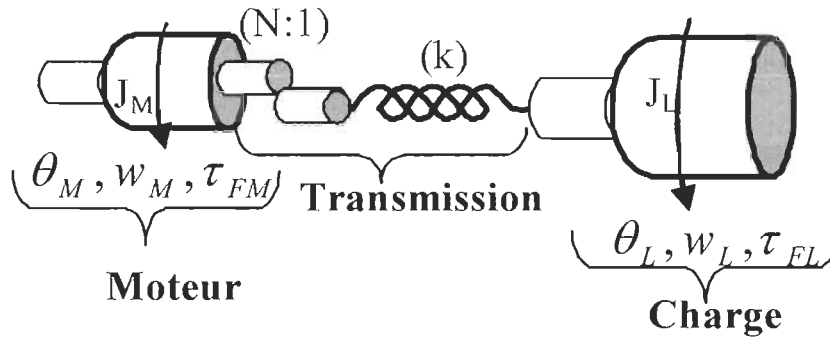


Figure 2.1 : Structure de l'articulation flexible

Les variables dépendantes du temps sont les positions, vitesses et accélérations angulaires respectivement notées θ , w et α , et les couples τ ; J représente l'inertie du moteur ou de la charge et sera considéré comme un paramètre *a priori* incertain; η est l'efficacité de la boîte de transmission; k la raideur du ressort de torsion équivalent à la boîte de transmission et N le rapport de transmission.

Les paramètres η , N et k seront supposés constants dans ce travail, mais ce n'est pas toujours le cas, notamment pour la constante de torsion.

Dans certains travaux de recherche, des modèles de flexibilité plus complexes ont été développés, mais le modèle retenu dans cette étude reste tout de même d'usage courant.

2.1.1 Modèles utilisés

2.1.1.1 Modélisation du moteur et de la charge

Les comportements du moteur et de la charge peuvent être décrits par des modèles similaires classiquement utilisés en commande. Ces modèles relient la position angulaire au couple appliqué, on y retrouve un double intégrateur et le modèle de friction comme sur la figure 2.2.

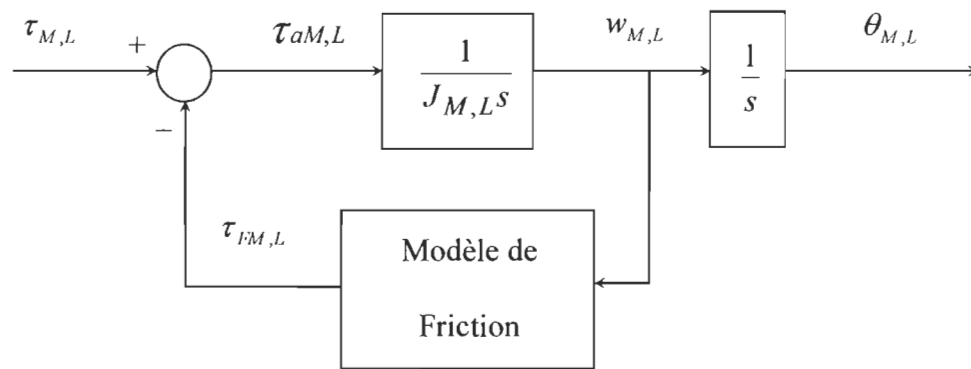


Figure 2.2 : Modèle du moteur et de la charge

Lorsqu'on applique un couple électromécanique au moteur, l'arbre de celui-ci effectue une rotation qui l'amène dans une certaine position angulaire. De même, un couple appliqué à la charge implique une rotation angulaire de celle-ci. Sur la figure 2.2 τ_M représente le couple électromécanique du moteur duquel est soustrait le couple de charge; τ_L représente le couple mécanique transmis à la charge.

2.1.1.2 Modélisation des phénomènes de friction

La conception d'une loi de commande pour un système automatisé est basée sur les critères de performance que le système doit satisfaire : rapidité de réponse, qualité de régulation, et surtout la stabilité et l'insensibilité face aux perturbations extérieures et autres effets indésirables. Parmi les effets indésirables auxquels sont soumis les systèmes électromécaniques, les phénomènes de friction peuvent fréquemment opérer une réduction non négligeable sur les performances et sur la durée de vie de ces systèmes. Il est donc très important de concevoir des lois de commande tenant compte de ces phénomènes et réduisant leurs conséquences.

Considérant les différents modèles de friction existants, l'utilisation de contrôleurs classiques tel que le PID s'avère insuffisante lorsque l'on considère les modèles les plus complets qui comprennent des termes fortement non linéaires.

Des travaux de recherche ont abouti à des moyens et des lois de commande plus complexes, pour apporter une compensation efficace au problème de friction. Parmi les solutions apportées, nous trouvons les modifications de lubrification, les modifications matérielles, les lois non modélisées (RNA, logique floue, etc.) et les lois de commande adaptative.

Nous allons présenter une brève revue des modèles des phénomènes de friction, le premier objectif étant de trouver une modélisation précise de la friction comprenant les forces de frottements statiques, visqueux et de Coulomb.

Les phénomènes de friction ou frottements ont un rôle important dans le contrôle des systèmes. La friction peut amener des erreurs de poursuite de trajectoire, des cycles limites et de nombreux autres effets indésirables [14]. Les stratégies de contrôle qui tentent de compenser efficacement ces phénomènes requièrent un modèle suffisamment exact.

De nombreux travaux de recherche ont été effectués dans ce but [14] [15] afin de proposer un modèle capable de prendre en compte tous les effets de friction connus à ce jour. Généralement, la force de friction totale est exprimée en fonction de la vitesse de glissement. Les modèles les plus simples de friction comprennent simplement les effets de frottements visqueux et de Coulomb comme sur la figure 2.3-a. On y trouve parfois une représentation simplifiée du frottement statique tel que sur la figure 2.3-b.

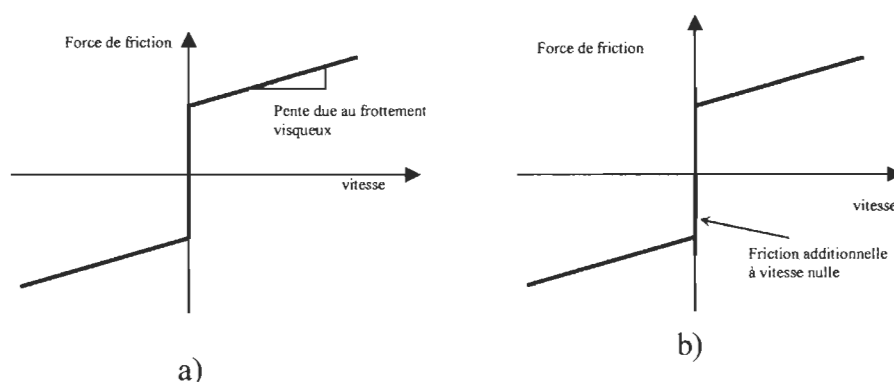


Figure 2.3 : Modèles simples de friction

Ces modèles sont largement utilisés mais ne permettent pas toutefois de capter tous les effets connus de friction et la qualité de compensation peut en être affectée.

Seidl utilise un modèle comprenant une bande morte autour de la vitesse nulle pour traduire l'effet de frottement statique [13].

Canudas de Wit propose un nouveau modèle en 1995 qui est une extension du modèle de Dahl et qui donne l'expression du modèle dynamique de friction sous forme d'une équation d'état dont la variable est la vitesse relative des surfaces de contact. Il donne également une méthode de compensation basée sur un observateur d'état mais qui suppose certaines connaissances *a priori* des phénomènes de friction considérés [14].

Finalement, le modèle qui paraît à l'heure actuelle proposer le meilleur compromis entre l'exactitude du modèle et les possibilités de compensation donne la force de friction instantanée au point de contact de deux surfaces en fonction de la vitesse instantanée de glissement entre ces deux surfaces. Ce modèle comprend trois termes décrivant les phénomènes de frottement de Coulomb, statique et visqueux [14]. Ainsi cette force peut s'écrire :

$$F_f(t) = F_c * \text{sign}(v(t)) + F_v * v(t) + F_s * \text{sign}(v(t)) * \exp\left(-\left(v(t)/v_s\right)^2\right) \quad (2.3)$$

Où F_c, F_v, F_s sont respectivement les coefficients de frottements de Coulomb, visqueux et statique, v_s est la vitesse caractéristique de frottement statique et $v(t)$ est la vitesse.

La figure 2.4 donne l'allure de cette fonction pour les valeurs des paramètres suivants :

$$\left\{ \begin{array}{lll} F_c = 0.0288 & \text{Nm} & (\text{ou N}) \\ F_v = 0.00055 & \text{Nm/(rad/s)} & (\text{ou N/(m/s)}) \\ F_s = 0.0191 & \text{Nm} & (\text{ou N}) \\ v_s = 0.95 & \text{rad/s} & (\text{ou m/s}) \end{array} \right.$$

Note : les unités entre parenthèses pour un axe en translation.

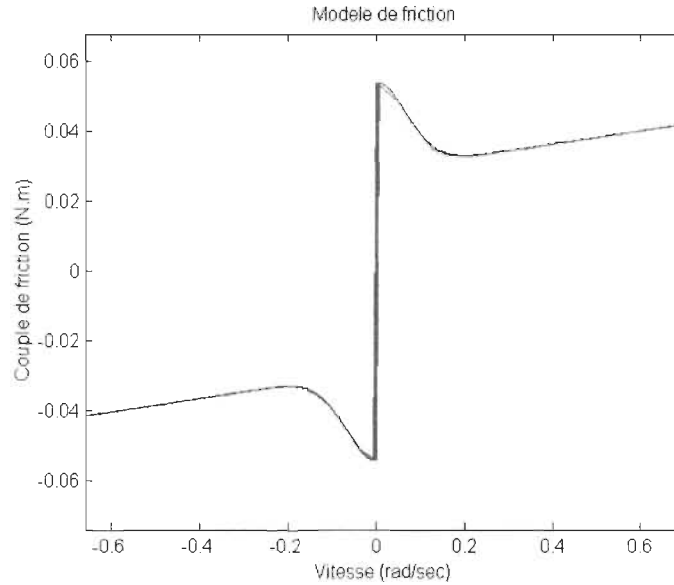


Figure 2.4 : Modèle de friction retenu

La non-linéarité autour de la vitesse nulle peut causer des oscillations numériques importantes. De plus, le couple de frottement ne doit pas entraîner la charge ou le moteur à l'arrêt ce qui serait le cas avec le modèle présenté par l'équation (2.2) et ne représenterait pas un comportement réel.

Pour palier à cela, le couple de frottements à très basse vitesse sera calculé différemment. On introduit une limite inférieure de vitesse ε en dessous de laquelle le couple de frottements arrêtera complètement le mouvement si le couple appliqué (couple total excepté les termes de frottements) au moteur ou à la charge est inférieur en valeur absolue au couple de frottements. En effet, si le couple appliqué est inférieur en valeur absolue au couple de frottements, le calcul du couple total appliqué à la charge ou au moteur devient négatif comme si le couple de frottements engendrait un mouvement.

L'algorithme qui suit permet de prendre cela en considération et de faire en sorte que les phénomènes de friction demeurent une contrainte sur le mouvement et non un phénomène générateur de mouvement (figure 2.5). Les variables utilisées sont celles du moteur ou de la charge, τ_f désigne le couple de frottement, τ le couple de commande, v la vitesse et ε est la limite de vitesse inférieure qui sera fixée arbitrairement la plus faible possible et de façon à éviter les oscillations numériques ou autres problèmes dus à la discontinuité des fonctions en 0.

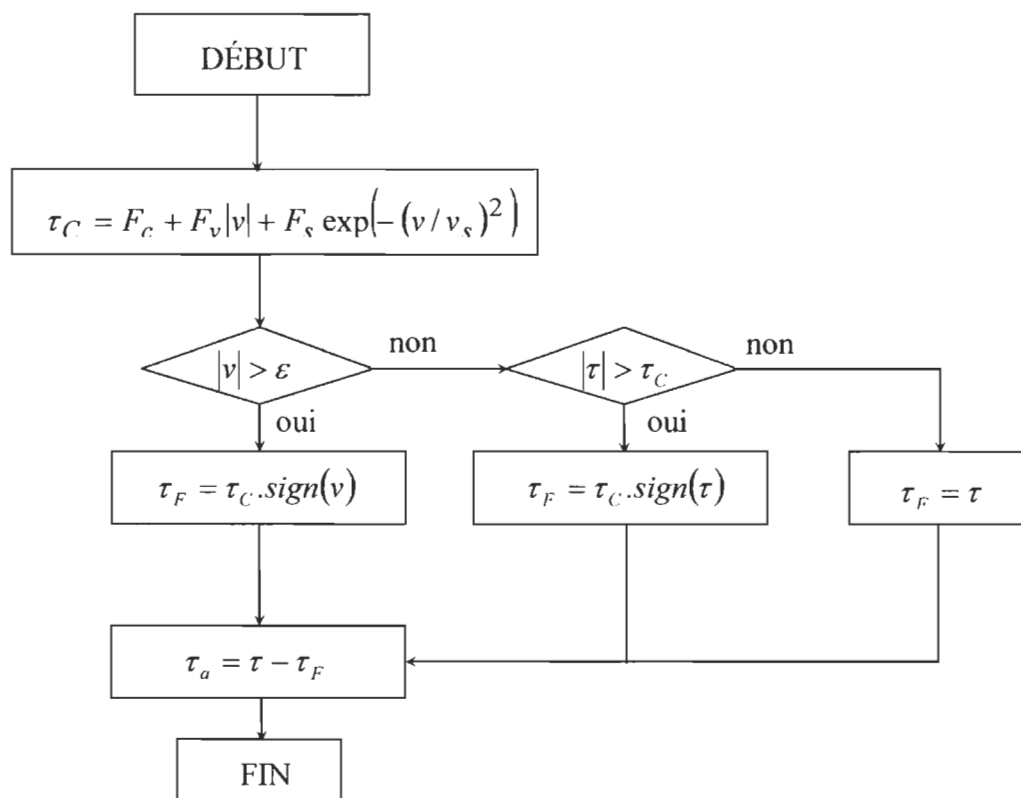


Figure 2.5 : Algorithme de calcul du couple de frottement

Cet algorithme sera utilisé pour modéliser les phénomènes de friction aussi bien du côté de la charge que du côté du moteur.

2.1.1.3 Modèle complet de l'articulation

Rappelons que nous considérons une articulation composée d'un moteur commandé en couple pilotant une charge, les deux étant reliés par une boîte de transmission équivalente à un ressort de torsion. Le moteur et la charge sont représentés par le modèle vu au paragraphe 2.1.1.1 et la boîte de transmission par une fonction linéaire de pente k représentant la raideur du ressort de torsion équivalent. Ainsi l'articulation peut être modélisée comme sur la figure 2.6. Ce modèle traduit bien l'équation (2.2).

Le problème de la flexibilité vient du fait que la boîte de transmission est équivalente à un ressort de torsion et ne permet pas d'isoler complètement le moteur de la charge dans le contrôle, cela est une difficulté notamment dans le placement des pôles permettant de trouver les gains du contrôleur proportionnel dérivatif. Une solution conservatrice consiste à considérer que le moteur et la charge sont séparables, i.e. négliger la flexibilité, puis tenter de supprimer ses effets en sur-amortissant le système ce qui permet d'éviter les problèmes causés par la flexibilité.

Ce type de transmission peut être la source de plusieurs phénomènes, par exemple le couple moteur n'est pas directement transmis à la charge, le mouvement de la charge peut créer un couple qui va être réfléchi vers le moteur, il peut y avoir apparition de cycles limites ou instabilité...

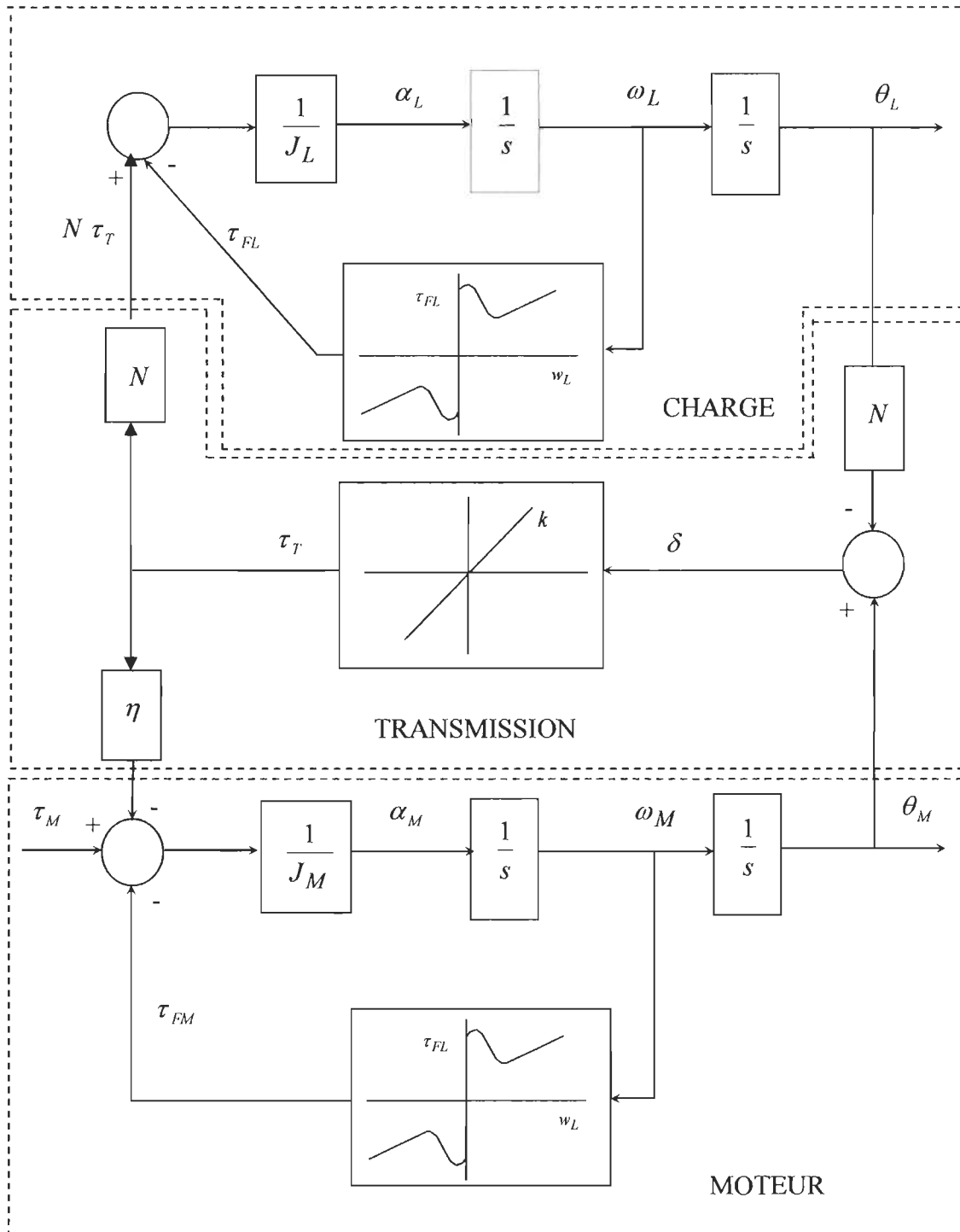


Figure 2.6 : Modélisation de l'articulation flexible

2.2 Commande inverse des systèmes

2.2.1 Modélisation inverse

Le concept d'inversion des systèmes a plusieurs applications dans les domaines de commande, filtrage et codage. Pour les systèmes linéaires avec une seule entrée et une seule sortie le problème d'inversion est trivial en utilisant le concept de fonction de transfert. Le modèle direct d'un système relie ses entrées à ses sorties (figure 2.7a). La modélisation inverse d'un système c'est de définir un modèle qui permet de trouver les entrées du système à partir de ses sorties, comme le représente la figure 2.7b, tel que si :

$$Y_d = F(u_d) \quad \text{alors} \quad u_d = F^{-1}(Y_d)$$

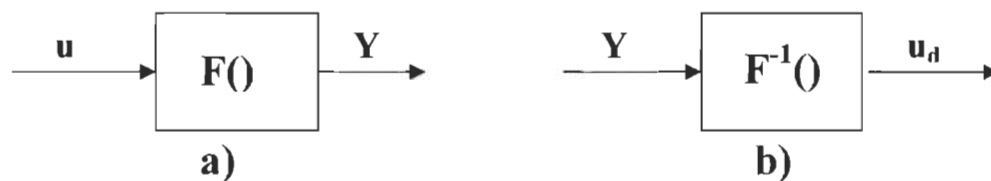


Figure 2.7 Modèles direct (a) et inverse (b).

Le système de commande est inversible quand la relation entrées-sorties est injective. Alors si on connaît la fonction de sortie, théoriquement on est capable de déterminer la fonction de commande appliquée.

Si l'on connaît la relation causale donnant la sortie en fonction de la commande, alors l'inversion de cette relation permet de calculer la commande nécessaire pour obtenir une sortie spécifiée à l'avance. La difficulté réside dans l'obtention d'une loi de commande réalisable. Le problème c'est de conserver la propriété de causalité lors de cette inversion (en continu, il faut conserver la propriété "puissance finie", c'est à dire conserver des propriétés passe-bas.). Dans notre modèle le couple de friction $\tau_F(\dot{\theta})$ est discontinu ce qui signifie que la fonction inverse n'est pas unique. Pour réaliser le modèle inverse nous ferons l'approximation de la fonction inverse avec un RNA qui est capable de faire l'approximation des fonctions non-linéaires.

2.2.2 Commande inverse

Le principe de commande inverse se résume à utiliser F^{-1} pour générer le signal d'anticipation de commande u_d à partir de la sortie désirée Y_d du système. L'exactitude de la fonction inverse, la présence de perturbations et la sensibilité du modèle inverse face aux variations de paramètres affectent directement la qualité du signal u_d appliqué au système initial F pour générer la sortie désirée Y_d (Figure 2.8).

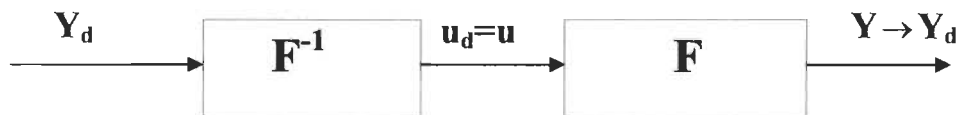


Figure 2.8 Commande inverse.

La technique présentée ici est très générale ; elle permet de commander un processus même non-linéaire, et est souvent employée en robotique pour inverser les relations causales dynamiques non-linéaires entre commande d'actionneurs et position d'un bras. Le modèle inverse permet à

partir d'une trajectoire désirée (la sortie) de trouver le couple électromagnétique qu'il faut appliquer (l'entrée).

Dans cette étude un réseau de neurones va apprendre le modèle inverse approximatif d'une articulation flexible dont le modèle a été développé dans ce chapitre.

2.3 CONCLUSION

Éliminer les effets de friction dans les machines actuelles devient un véritable défi. Pour cela, tous les aspects de la conception des systèmes, comprenant les concepts matériels tels que la lubrification et aussi les possibilités de contrôle, doivent être considérés et modélisés. Ainsi, de nombreuses techniques de compensation de friction et de flexion ont été développées, méthodes généralement développées empiriquement puisqu'il est encore difficile d'obtenir des modèles exacts pour ces phénomènes.

Aujourd'hui, les phénomènes de flexion et friction, présents dans les systèmes automatisés, sont mieux compris, et des modèles de plus en plus précis de ces phénomènes comprenant notamment la dynamique de friction sont disponibles.

Dans cette étude, nous allons utiliser le modèle de l'articulation flexible présenté dans ce chapitre pour développer l'algorithme de commande.

Chapitre 3

Commande des systèmes électromécaniques

3.1 Problématiques reliés à la commande des systèmes électromécaniques.

Les performances des systèmes électromécaniques sont limitées par les propriétés des éléments mécaniques qui les constituent. Les performances dynamiques des systèmes sont fortement affectées par la flexibilité structurelle et par les non-linéarités dures des phénomènes de frottement entre autres. L'écriture d'une loi de commande efficace pour de tels systèmes doit être réalisée en fonction de toutes ces contraintes.

3.1.1 Compensation de friction.

Une première difficulté associée aux modèles étudiés est de pouvoir identifier les paramètres de friction pour un système donné. Dans les modèles que nous avons évoqué, nous retrouvons systématiquement les paramètres F_c, F_v, F_s, v_s caractérisant les frottements de Coulomb, visqueux et statiques. Les paramètres F_v et F_c peuvent être identifiés par expérimentation avec quelques points de mesure à vitesse constante. Les autres paramètres sont plus difficiles à identifier. Si ces paramètres étaient constants, il serait très facile de compenser la friction en calculant en continu un terme de compensation d'amplitude égale au terme de friction mais de signe opposé.

Cependant, ces paramètres ne sont pas constants, ils peuvent souvent varier suivant l'évolution du système. Ceci justifiera notamment l'emploi de lois de commande adaptative pour la compensation puisque ces méthodes intègrent une identification des paramètres suivant le comportement dynamique du système.

La deuxième difficulté importante est liée à la simulation numérique du comportement d'un système comprenant des phénomènes de friction. En effet, nous avons pu constater qu'il y avait une discontinuité de la courbe de friction à vitesse nulle, et ceci pose un problème aux intégrateurs numériques. Ce phénomène affecte aussi les performances en commande à cause de la non unicité du modèle inverse de l'articulation en ce point.

Les figures 3.2 à 3.4 permettent d'observer les effets de l'ajout de la friction au système (figure 3.1) qui répond à l'équation dynamique suivante [16] :

$$M \frac{dv}{dt} = u(t) - F_f(t)$$

où M est la masse du système

$u(t)$ est la force instantanée (ou couple) appliquée à la masse, friction exceptée.

$F_f(t)$ est la force instantanée totale de friction. Cette force totale de friction peut aussi s'écrire :

$$F_f(t) = F_c * \text{sign}(v(t)) + F_v * v(t) + F_s * \text{sign}(v(t)) * \exp\left(-\left(v(t)/v_s\right)^2\right)$$

Cette force représente le modèle de friction retenu dans cette étude.

Les paramètres Kp et Kd du correcteur P+D, ont été fixés pour rencontrer les critères suivants :

- dépassement nul.
- erreur stationnaire nulle.
- temps de stabilisation désiré.

Le terme de friction peut être calculé à partir de la vitesse précédente mesurée ou aussi à partir d'un estimé de la vitesse.

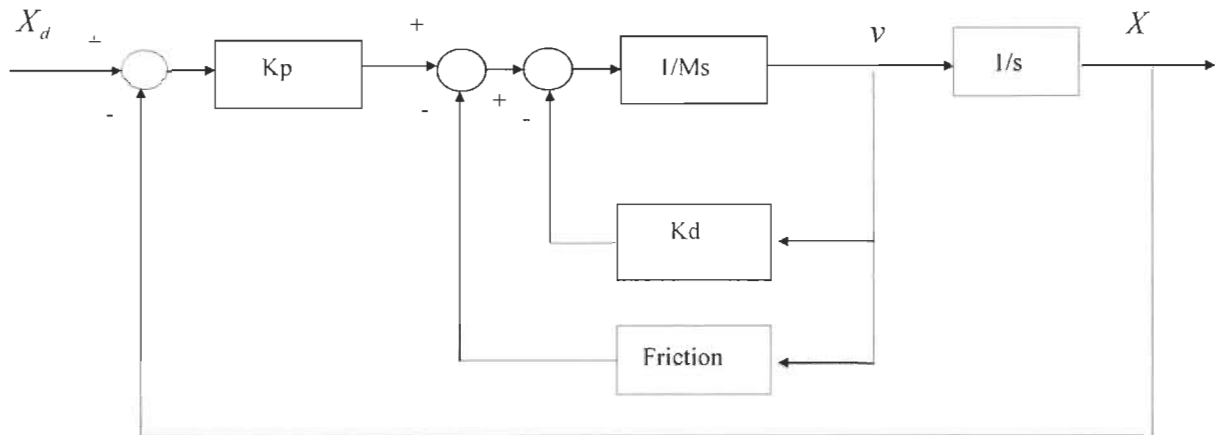


Figure 3.1 : Système comprenant le module de friction.

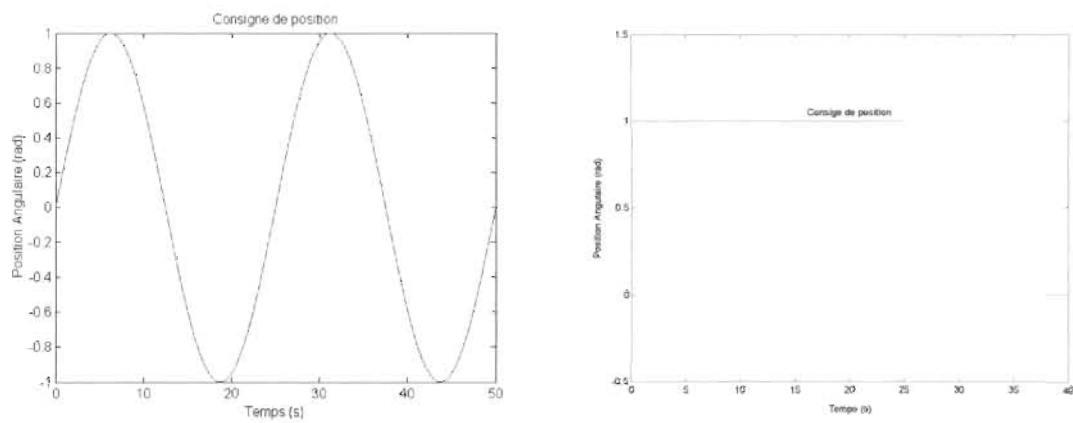


Figure 3.2 : Consignes de position

Réponse indicielle :

Sur la réponse indicielle du système avec friction (Figure 3.3), on peut constater qu'à haute vitesse la friction a peu d'effet sur la forme de la réponse, en revanche à vitesse faible, elle tente d'arrêter le mouvement et on observe ainsi une nette atténuation par rapport à la valeur de consigne, aussi bien en valeur maximale que minimale.

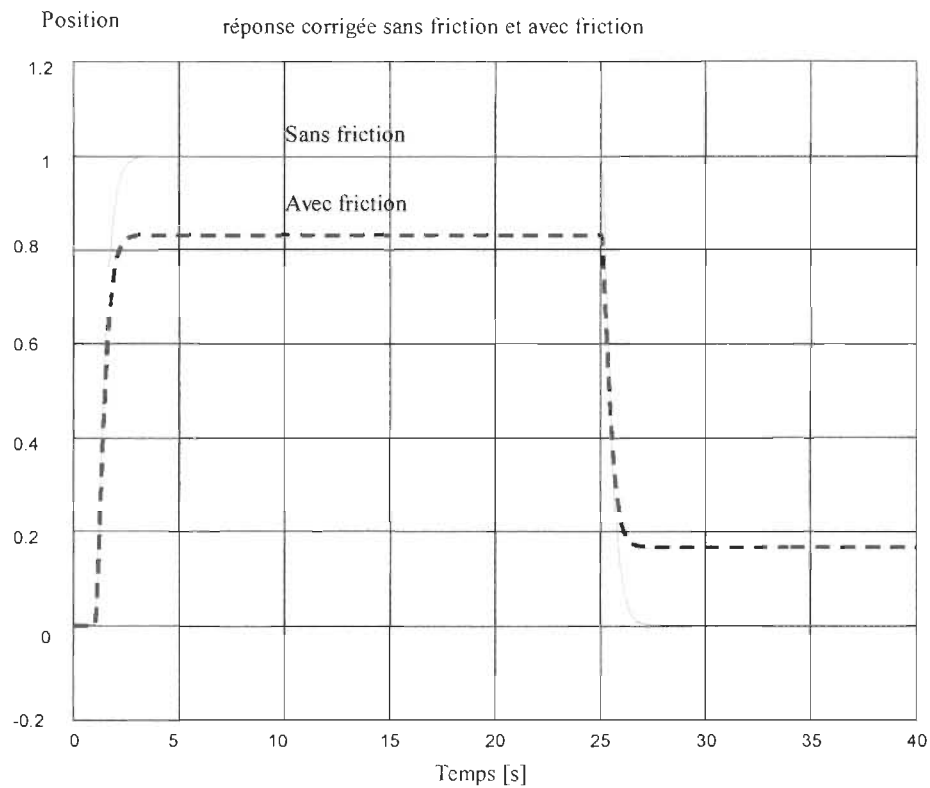


Figure 3.3 Effets de la friction sur la réponse indicielle.

Réponse sinusoïdale :

Les mêmes effets sont observables sur la réponse sinusoïdale (Figure 3.4). L'effet de la friction à vitesse plus faible est encore plus visible car on le voit déjà sur les phases de montée et de descente, la pente (vitesse) y étant moins élevée que dans le cas de l'échelon.

On peut constater que la friction ralentit la réponse du système et que celui-ci a du mal à quitter les phases de vitesse faible, là où la friction s'impose plus facilement devant le couple qui génère le mouvement.

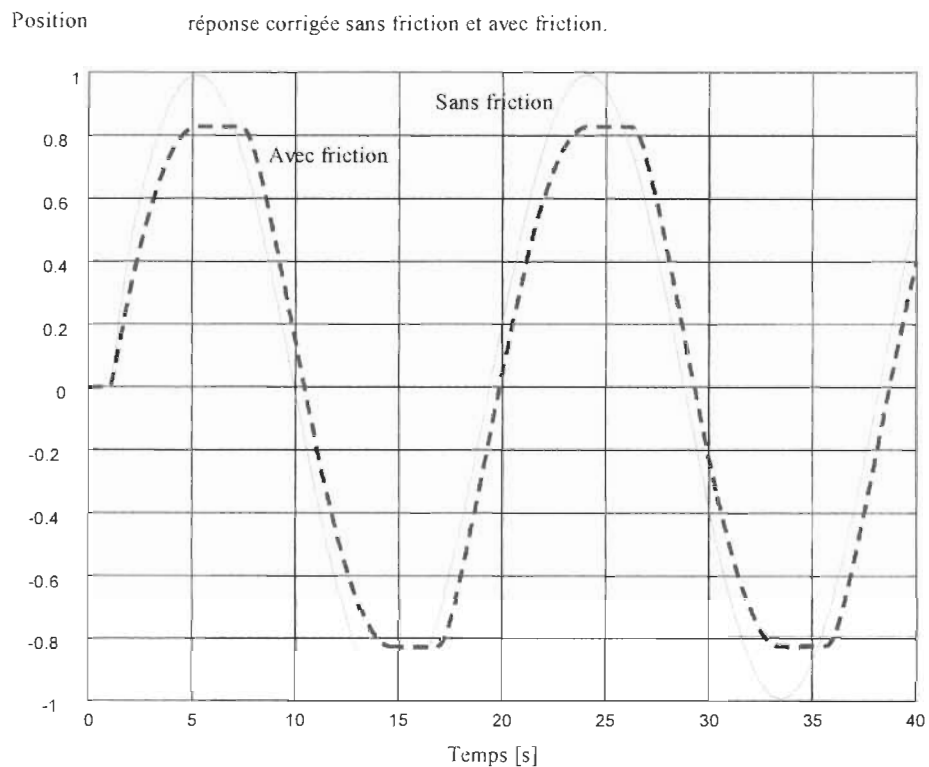


Figure 3.4 Effets de la friction sur la réponse sinusoïdale.

L'utilisation d'une action intégrale simple pour compenser la friction causera des oscillations sous forme de cycle limite. Les méthodes pratiques peuvent être divisées en trois groupes : l'interdiction de friction, les techniques non modélisées et les techniques modélisées.

1. L'interdiction de friction

On désigne par cette appellation les techniques permettant d'éliminer la friction ou plutôt de la limiter considérablement. Elles passent en général par des modifications apportées au système, modifications pouvant porter sur la lubrification (type et quantité) ou sur des propriétés physiques du système (réduction de masse, etc.)

2. Techniques non modélisées

Il s'agit en général d'utiliser des techniques correctrices non basées sur des modèles. On cite par exemple les modifications sur le contrôle intégral, l'utilisation de gains élevés, l'apprentissage, le contrôle de couple, etc.

3. Techniques modélisées

Quand la friction peut être prédite avec suffisamment de précision, elle peut être compensée par rétroaction d'une force égale en amplitude à la force totale de friction et de signe opposé. La quantité de friction présente dans le système varie avec le temps à cause de certains facteurs environnementaux tels que la température ou les conditions de lubrification. C'est pourquoi les méthodes pratiques de compensation de friction

impliquent une identification en ligne des paramètres. Ces méthodes sont dites adaptatives et sont soit directes ou indirectes :

a. Commande adaptative directe :

Dans les lois de commandes adaptatives directes, il n'y a pas d'identification propre des paramètres de friction, mais plutôt des gains de correction visant à limiter l'erreur de poursuite.

b. Commande adaptative indirecte :

Ce type de commande comprend une identification en ligne des paramètres de friction. Il présente notamment l'avantage d'avoir une grande qualité des paramètres estimés puisque ceux-ci peuvent être vérifiés avant d'être utilisés dans la boucle de compensation, comme il est proposé par Canudas de Wit [17], par exemple, où le modèle de friction est écrit

$$\text{comme suit : } F_f(t) = \Theta^T * \Omega(t)$$

où Θ est le vecteur des paramètres de friction et $\Omega(t)$ est le vecteur de régression qui dépend du système et peut contenir des fonctions non linéaires suivant le modèle de friction considéré :

$$\Theta^T = [F_c; F_s; F_v]$$

$$\Omega(t) = \left[\text{sign}(v(t)); \frac{1}{1 + (v(t)/v_s)^2}; v(t) \right]$$

Pour le système dynamique :

$$M * \frac{dv(t)}{dt} = u(t) - F_f(t)$$

En mode échantillonné, l'erreur de prédiction à $t = k * T$ est donnée par :

$$e_k = u_k - M * \left. \frac{dv(t)}{dt} \right|_k - \hat{\Theta}^T * \Omega_k$$

$\hat{\Theta}_k$ représente le vecteur estimé des paramètres de friction à l'instant $t = k * T$

Ce vecteur de paramètres est calculé par un algorithme qui donne la relation entre $\hat{\Theta}_k$ et

$\hat{\Theta}_{k-1}$:

$$\hat{\Theta}_k = \hat{\Theta}_{k-1} + \lambda_k * P_k * \Omega_k * e_k$$

où λ_k est le gain de l'algorithme et P_k est la matrice identité.

La nouvelle équation du système compensé est:

$$M * \frac{dv(t)}{dt} = u(t) - F_f(t) + \hat{\Theta}^T * \Omega(t)$$

3.1.2 Compensation de flexion.

Les systèmes d'entraînement qui emploient des boîtes d'engrenage ou un élément de transmission de couple quelconque, les systèmes d'entraînement hydrauliques et les manipulateurs de grande dimension ont une flexibilité inhérente au système d'entraînement ou à la structure qui est un facteur limitatif important des performances du système. Ceci implique que, entre autres, les lois de commande pour les manipulateurs dits rigides ne s'appliquent pas directement aux manipulateurs flexibles. D'autre part la flexibilité ajoute des modes qui peuvent être excités par des forces externes. Plusieurs lois de commande ont été proposées pour la commande de manipulateurs avec joints flexibles. Parmi les approches proposées, mentionnons les techniques de :

- Perturbations singulières [2] qui utilisent une forme de contrôle basé sur la décomposition du système en deux sous-systèmes selon leur échelle de temps (c.-

à-d. deux boucles indépendantes, une pour le sous-système lent et l'autre pour le sous-système rapide).

- La linéarisation par rétroaction [2] qui demande une connaissance exacte des paramètres du modèle et un retour complet de l'état afin de linéariser le modèle.
- La reconstruction des entrées [12] où on utilise des filtres causals pour ne pas exciter les fréquences naturelles du système en générant des signaux de références souples.
- La conception basée sur la passivité [12] qui est définie comme une condition reliant l'entrée et la sortie pour une classe de systèmes physiques qui ne génèrent pas d'énergie.

Ces travaux considèrent un modèle qui exclut toutes les non-linéarités dures (frottement sec, retour de dents, etc.), tel que ces résultats demeurent souvent académiques.

3.2 Méthodes modernes de commande pour les systèmes non-linéaires .

On a trouvé que la friction cause des erreurs significatives en commande de position. Les méthodes de compensation basées sur le modèle, dans les systèmes électromécaniques avec plusieurs actionneurs, sont trop complexes et dépendent des paramètres qui varient dans le temps.

Les méthodes de commande intelligentes et adaptatives peuvent résoudre ces problèmes, en utilisant les réseaux de neurones artificiels (RNA) et/ou la logique floue.

3.2.1 Réseaux de neurones artificiels (RNA).

Depuis quelques années, les recherches sur la commande et l'automatisation industrielle se sont multipliées. Différentes méthodes avancées de traitement numérique ont été utilisées afin de proposer des solutions à la problématique de commander des systèmes non-linéaires. Une technique qui est de plus en plus populaire consiste à tirer profit des algorithmes basés sur les réseaux de neurones artificiels (RNA). Les RNA offrent des caractéristiques intéressantes autant au niveau algorithmique qu'au niveau de leur mise en œuvre dans une technologie ITGE [18].

On présente le RNA comme un nouvel outil dans le domaine de traitement de l'information. Le concept de RNA est né d'un essai de reproduire le système nerveux humain. Comme le cerveau est composé des cellules neuronales reliées à grande échelle, les RNA, les modèles de cerveau, sont composés des modèles linéaires et non linéaires de cellules neuronales qui traitent les informations d'une façon distribuée et parallèle en contraste de la structure des ordinateurs conventionnels de Von Neumann où la vitesse est limitée par le délai de propagation dans les transistors. À cause de sa structure massivement parallèle le réseau effectue les calculs avec une vitesse supérieure avec plusieurs autres caractéristiques intéressantes dont la capacité d'apprentissage et d'auto organisation. Ils peuvent aussi s'adapter aux changements des données.

Dans cette section, nous ferons tout d'abord un survol des fondements des RNA en citant les principaux types de neurones et de topologies de RNA. Ensuite, nous discuterons des

propriétés des RNA qui les rendent intéressants pour résoudre la problématique de commande des systèmes non-linéaires. Enfin, nous passerons à la pièce maîtresse de ce mémoire, les RNA multicouches. Nous en ferons la définition et la description.

3.2.1.1 Principes de base des RNA.

Les RNA s'inspirent du cerveau humain. Les neurones qui constituent les centres de décisions sont reliés de façon à accomplir un travail bien défini. Chaque neurone reçoit des stimuli qui entraînent la prise d'une décision en relation avec le processus qui lui est propre. Les RNA constituent une méthode de programmation qui consiste à établir les paramètres de chaque neurone ainsi que les relations entre chacun d'eux. En établissant les paramètres optimums, nous pouvons faire l'approximation d'une fonction de façon précise. Les réseaux de neurones représentent une base algorithmique puissante pour le traitement de signaux grâce à leurs propriétés, principalement la non-linéarité et le parallélisme intrinsèque à la méthode.

3.2.1.1.1 Neurone formel.

Tous les types de neurones ont le même principe de base. On associe à chaque entrée un poids w , c'est à dire un facteur multiplicatif. Pour garder l'analogie avec le cerveau humain, l'adaptation des poids du RNA est appelée apprentissage. Les produits sont ensuite sommés pour constituer la base du processus de décision. C'est ce que nous appellerons la somme des produits. Par la suite, le résultat passe dans une fonction de

décision $g(\bullet)$ afin d'en évaluer sa valeur de sortie. La Figure 3.5 montre le neurone formel.

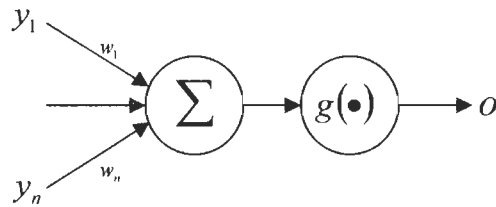


Figure 3.5 Neurone formel

3.2.1.2 Topologies de RNA.

Les RNA sont des ensembles de neurones reliés, maillés. Ils sont divisés en couches qui sont constituées de neurones de même type. Des couches de différents types peuvent constituer le RNA pour plus de flexibilité afin de répondre à un plus grand nombre d'applications. La schématisation d'un RNA multicouche est représentée à la figure 3.6.

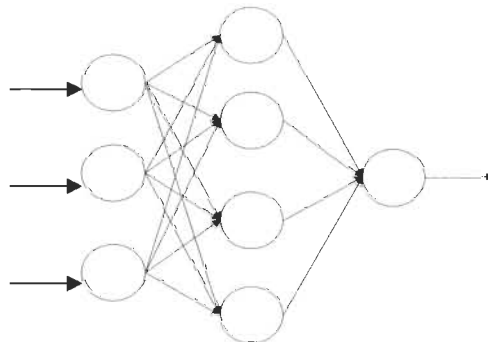


Figure 3.6 Schématisation d'un RNA multicouche

Le nombre de couches change selon la complexité des données à apprendre, le nombre d'entrées et le nombre de sorties du RNA. La couche de transmission est celle qui transmet les entrées au RNA. Elle peut ou non avoir une influence sur la valeur de sortie. Il y a un neurone par entrée. La couche de sortie est celle qui produit la sortie du RNA. Il y a un neurone par sortie. Les couches cachées se situent entre la couche de transmission et la couche de sortie. On les appelle ainsi puisqu'elles sont invisibles de l'entrée et de la sortie du RNA.

Le premier modèle est appelé neurone à seuil. Ce neurone est le plus simple, car sa sortie est constituée d'un seul bit. La fonction de décision est la fonction unitaire. Si l'entrée de la fonction est inférieure à 0, la sortie prend la valeur « 0 ». Dans le cas contraire, la sortie prend la valeur « 1 » tel que montré à la Figure 3.7 a. Ce type de neurone est souvent utilisé pour des variables linéairement séparables. Des variables sont linéairement séparables si dans le plan, les « 0 » peuvent être séparés des « 1 » à l'aide d'une ou plusieurs droites. À la figure 3.8 on retrouve sur les axes les deux entrées alors que les « 0 » et les « + » représentent la sortie (0 ou 1). On peut voir que les variables sont séparées à l'aide d'une seule droite donc le réseau a seulement un neurone à seuil. À cause de cette limitation, on a introduit les réseaux multicouches qui sont aptes à séparer des variables à l'aide de plusieurs droites.

Le second neurone est du type « Adaline » (ADAPtive Linear Neuron) ou plus simplement neurone linéaire dont la fonction d'activation est présenté à la Figure 3.7 b. La sortie du neurone est directement la somme des produits. La sortie est évidemment un

nombre réel. Ce neurone est souvent retrouvé à la sortie d'un réseau Perceptron multicouche qui utilise des neurones de forme sigmoïde sur la couche cachée.

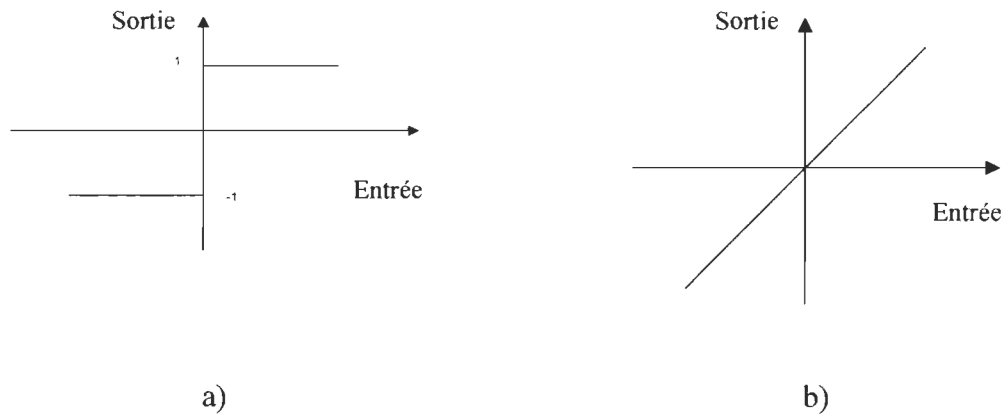


Figure 3.7 Fonction d'activation du neurone à seuil (a) et du neurone linéaire (b)

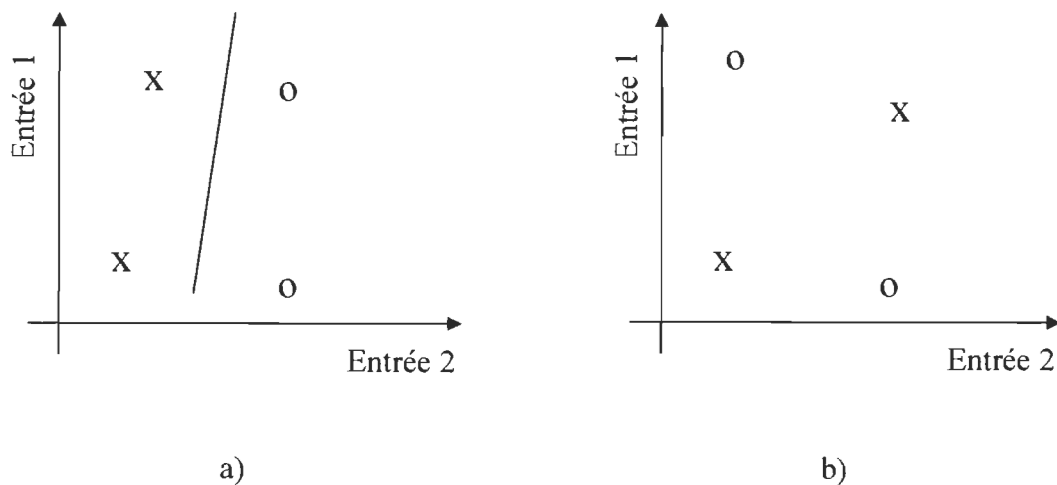


Figure 3.8 Variables linéairement séparables (a) et linéairement non séparables (b)

Le neurone de sigmoïde est le plus complexe de tous les neurones vus jusqu'à maintenant mais il permet l'apprentissage d'un éventail plus élaboré de fonction. La fonction de décision de celui-ci est la fonction tangente hyperbolique. Comme les entrées, les poids et le seuil sont des nombres réels, la sortie sera également un nombre réel qui varie entre 0

et 1. L'application d'une fonction linéaire par morceau permet d'implanter des approximations des fonctions sigmoïdes de façon compacte. La figure 3.9 montre un exemple de deux fonctions d'activation.

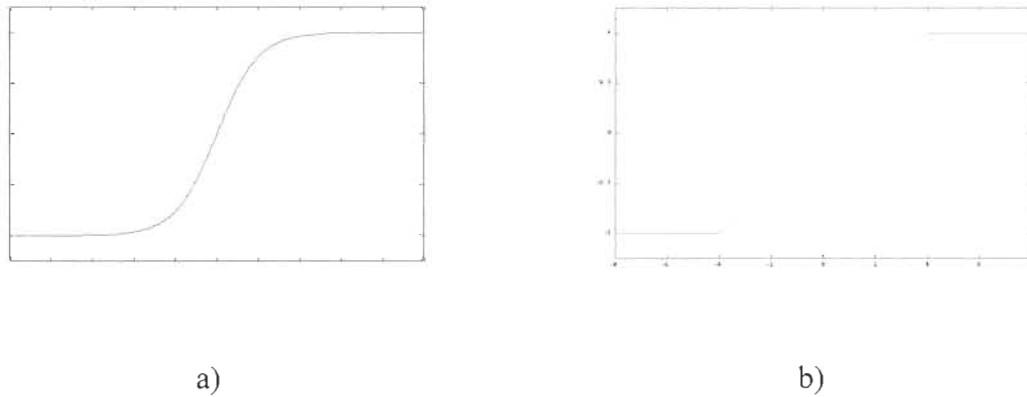


Figure 3.9 Fonctions d'activation, a) sigmoïde et b) linéaire par partie « piecewise linear »

Les topologies de RNA pour les applications de commande sont vraiment variées. Les connections, le nombre de couches cachées et les fonctions d'activation peuvent être modifiés pour obtenir un nouveau RNA pour en accroître les performances. Principalement, nous pouvons les classer en deux grandes catégories: les RNA multicouches et les RNA récurrents.

3.2.1.2.1 Les RNA récurrents.

Une topologie de RNA qui tient compte de la dynamique des données est le modèle NARX tel que montré à la figure 3.10. Le RNA NARX s'inspire du RNA multicouche, présenté à la figure 3.11. Une rétroaction de la sortie du réseau est introduite ainsi qu'un

nombre déterminé de retards sur celle-ci et sur les valeurs en entrée. Ainsi, on a la possibilité d'utiliser le modèle ARMA bien connu sous forme de RN. L'avantage que le RN a face aux autres méthodes de ce type, est qu'il n'a aucune valeur à syntoniser mis à part le pas d'apprentissage. Également, la connaissance du procédé n'est pas requise.

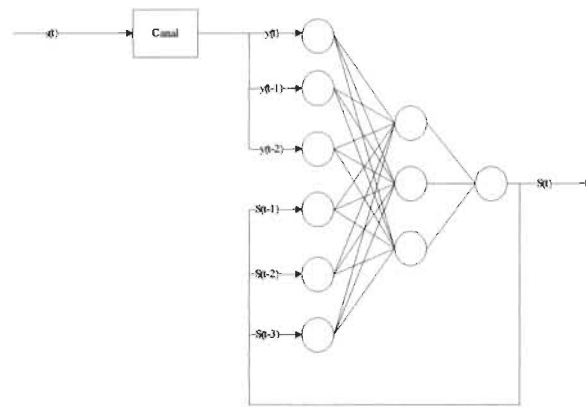


Figure 3.10 Schématisation d'un RNA récurrent (NARX).

3.2.1.2.2 Les RNA multicouches.

Plusieurs propositions de RNA multicouches ont été faites au cours des années. Étant une évolution du neurone formel de base, les RNA multicouches sont venus régler bien des problèmes de classification et d'estimation. On a même dit qu'ils sont des estimateurs universels. Donc plusieurs auteurs les ont repris afin de résoudre le problème de modélisation.

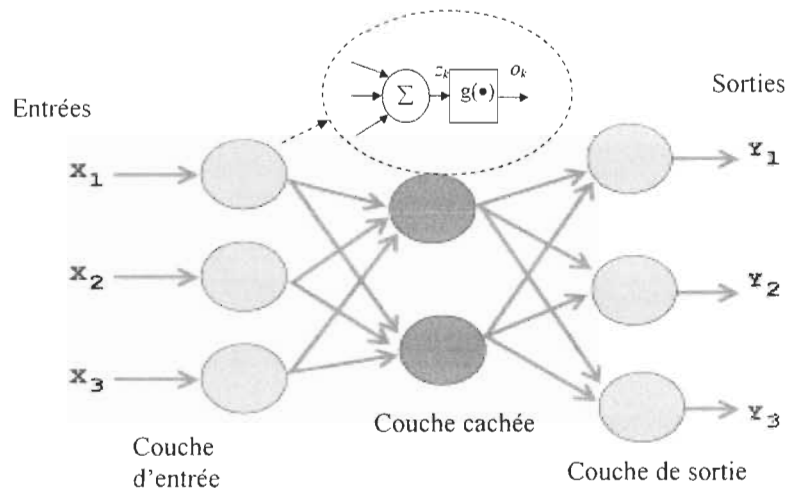


Figure 3.11 Réseau de neurones multicouche.

La comparaison de deux RNA multicouches, le premier comportant des connections régulières et le second des connections irrégulières, a été faite et il a été démontré qu'en ayant des connaissances *a priori* sur le système, le deuxième permet d'obtenir de meilleurs résultats. Cependant, l'absence de récurrence rend ce type de RNA moins robuste mais ils sont très attrayants pour une intégration à très grande échelle puisqu'il n'y a pas de dépendance dans le réseau.

3.2.1.2.3 RNA Hopfield.

Les RNA entièrement connectés, tel le réseau d'Hopfield, ont prouvé leur valeur face au problème de classification. Les récurrences dans le réseau permettent une interpolation efficace. Par exemple, ce type de réseau peut servir de banque d'images. Lorsqu'on lui fournit une image incomplète, il est apte à retrouver l'image originale. Il est à noter qu'il

n'est pas nécessaire d'avoir une entrée et une sortie par neurone. Lorsque la fonction d'activation utilisée est une fonction seuil, alors il s'agit du réseau d'Hopfield tel que montré à la Figure 3.12.

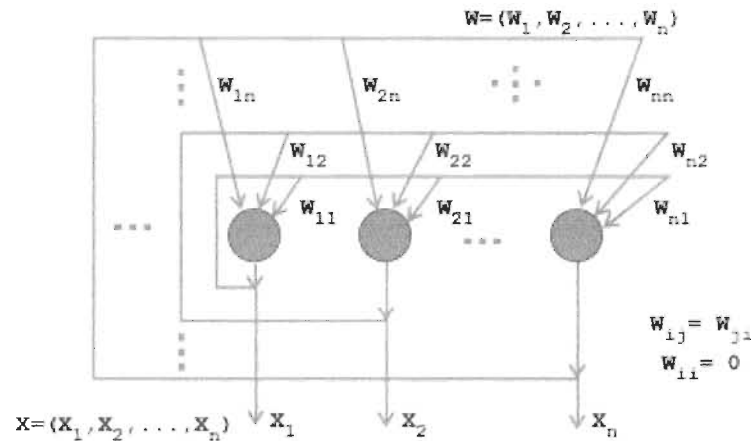


Figure 3.12 Réseau Hopfield.

La flexibilité obtenue avec de tels modèles et topologies permet une grande variété d'applications. Le réseau peut rester intact alors que nous varions l'apprentissage pour une autre affectation, que ce soit en traitement numérique des signaux, contrôle ou autre. Ici, nous évaluerons s'il est possible d'appliquer les RNA au problème de commande d'une classe de systèmes non-linéaires.

3.2.1.3 RNA pour la commande.

Les réseaux de neurone donnent avec un apprentissage ou un entraînement approprié une commande plus intelligente face à la dynamique non linéaire ou inconnue et les paramètres variants de l'environnement.

Les caractéristiques désirables des RNA pour la commande sont :

- 1- la capacité d'apprendre les données d'entraînement ;
- 2- le traitement parallèle des données ;
- 3- la robustesse face aux bruits ;
- 4- la capacité de généralisation, c'est-à-dire qu'ils peuvent interpoler entre les points de l'apprentissage pour fournir une réponse avec une grande précision ;
- 5- la capacité de faire l'approximation de fonctions non-linéaires.

Un RNA multicouches régulier sera utilisé dans l'algorithme de commande élaboré.

3.2.2 Commande par mode de glissement.

La théorie de la commande par mode de glissement (MG) couvre les systèmes à entrées et sorties simples ou multiples, les systèmes monovariante et multivariables, les systèmes linéaires et non linéaires, ainsi que les systèmes d'ordre élevé.

Un système à structure variable (SSV) est constitué d'un groupe de sous systèmes continus et d'une logique de sélection. On peut ainsi combiner les propriétés utiles de chacune des structures ou encore donner au système des propriétés non présentes dans ces structures. Dans la commande en MG, le point représentatif du système (point défini par l'ensemble des variables d'état du système) est contraint à se déplacer le long d'un ou plusieurs hyperplans dans l'espace d'état. De cette façon, l'insensibilité face aux variations

de paramètres et le rejet des perturbations sont assurés. Cependant, pour obtenir un tel régime de glissement, la loi de commande doit être discontinue [4].

Dans la littérature, on attribue les caractéristiques suivantes aux SSV fonctionnant en MG

1. grande précision ;
2. réponse dynamique rapide et sans dépassement (si désiré) ;
3. simplicité de conception et d'implantation ;
4. robustesse ;
5. le problème de stabilité globale du système n'existe pas : la loi de commande est définie de façon à ce que toutes les trajectoires se dirigent vers les hyperplans de glissement; il suffit alors que ces hyperplans de glissement définissent un système stable ;
6. la connaissance des valeurs des paramètres du système n'est pas nécessaire pour la conception : seules les valeurs limites des paramètres sont requises ;
7. les dérivées de la variable à commander peuvent être utilisées pour accélérer la réponse et permettre une plus grande variation des paramètres ; cependant, ceci diminue l'immunité du système face au bruit ;
8. les non linéarités et les interactions (couplages) peuvent être traitées et rejetées comme des perturbations ;
9. invariance face aux variations de paramètres: conception plus souple, précision des calculs moins importante ;

10. la performance d'un système de commande par MG est insensible à la présence de faibles constantes de temps dans le système tels le délai de commande dans l'élément de puissance et les retards introduits par les filtres dans les retours. Cependant, ces constantes de temps peuvent causer des oscillations à haute fréquence.

Malgré le fait que le réglage par mode de glissement possède ces avantages incontestables, cette méthode a été peu appliquée à cause des désavantages comme la forte oscillation de l'organe de commande et la nécessité d'utiliser la grandeur à régler et un certain nombre de ses dérivées, selon l'ordre du système. De plus, il y a une phase de convergence pendant laquelle la trajectoire, à partir d'une condition initiale quelconque hors de l'hyperplan de glissement, tend vers celui-ci. Pendant cette phase, le système est sensible aux variations de paramètres et aux perturbations. Cependant, de nombreux travaux ont permis de développer différentes stratégies pour corriger ces faiblesses.

- Utilisation des dérivées de la variable à régler :

Les dérivées de la variable à régler sont fréquemment requises pour réaliser une commande convenable. Ceci limite grandement le champ d'application des correcteurs utilisant le MG. Cependant, des méthodes indirectes de définition de la trajectoire ont été développées, méthodes qui permettent l'utilisation d'un nombre réduit de dérivées de la variable à régler. Cependant, dans le cas de la commande de position d'une articulation robotique, la vitesse est obtenue aisément et l'accélération peut être estimée. Un filtre peut alors être requis. Ce problème a donc peu d'influence dans le cas étudié.

- Phase de convergence :

Pendant la phase de convergence de la trajectoire réelle vers l'hyperplan de glissement, le système est sensible aux variations de paramètres et de la charge. Cette phase doit donc être minimisée ou éliminée.

- Activité intense du signal de commande :

La commande par correcteur fonctionnant en MG présente des discontinuités sur les hyperplans de glissement (plans de commutation) qui causent des oscillations (*chattering*) à la sortie du système et une activité intense du signal de commande. Ces deux phénomènes sont amplifiés par la présence de faibles délais (fréquence de commutation limitée) et de faibles constantes de temps négligées lors de la conception.

Ainsi, alors que la commande en MG résulte en une loi de commande robuste face aux variations de paramètres et aux perturbations, elle n'est pas robuste face aux approximations usuelles de modélisation (les modes négligés qui résident hors de la plage de fréquences d'intérêt). Ainsi, l'effet de la commutation constante de la commande sur les éléments mécaniques du système doit être considéré. De plus, l'instabilité du signal de commande peut causer une erreur stationnaire à la sortie.

Ainsi, le signal de commande doit être traité afin d'atténuer ou d'éliminer les oscillations.

Différentes stratégies ont été proposées dans la littérature afin d'éliminer les oscillations :

- (a) ajout d'un filtre passe bas à l'entrée du procédé ;
- (b) régulation de la dérivée de l'erreur par rapport à l'hyperplan de glissement ;
- (c) substitution du terme discontinu de la loi de commande par une fonction d'adoucissement. Cette fonction d'adoucissement est définie par :

- 1- une approximation continue du terme discontinu ;
- 2- une bande de largeur constante avoisinant l'hyperplan de glissement à l'intérieur de laquelle le gain est variable ;
- 3- une bande de largeur variable avoisinant l'hyperplan de glissement à l'intérieur de laquelle le gain est variable.

Avec toutes ces méthodes, l'élimination des oscillations est obtenue au coût d'une perte de robustesse face aux spécifications dynamiques et d'une complexité de conception et d'analyse accrue. Cependant, avec l'utilisation d'une bande d'adoucissement, la convergence du système vers cette bande est assurée. La perte de performance est donc plus aisément quantifiable. Notons que la méthode utilisant la régulation de la dérivée de l'erreur par rapport à l'hyperplan de glissement est très efficace mais que cette méthode utilise une adaptation du gain pour effectuer cette régulation. Cette adaptation peut être complexe et causer des problèmes de stabilité locale.

La commande par régulateur à structure variable fonctionnant en mode de glissement présente de nombreux avantages dont la robustesse, la simplicité et la possibilité de commander les systèmes variants dans le temps ou incertains et les systèmes non linéaires. Cette approche sera donc employée dans le développement du contrôleur.

3.2.3 Logique floue.

Le concept de la logique floue a été introduit en 1965 par Lotfi Zadeh, un professeur d'électronique à l'université de Berkeley (USA). Sa "Fuzzy Set Theory" n'eut pas un succès immédiat. Elle a été développée surtout en Europe et au Japon. Ce dernier, poussé par une pénurie de programmeurs, était à la recherche de méthodes pouvant réduire les coûts de développement des logiciels. Freinée par la réticence face à un nouveau concept et par des difficultés de réalisation pratique, la logique floue ne débouche réellement qu'en 1990 avec l'apparition de produits et de systèmes utilisant cette technique.

Aujourd'hui, la logique floue est arrivée à maturité et est utilisée dans de nombreux produits "grand public" notamment au Japon. Sa mise en oeuvre est maintenant facilitée par la disponibilité de microprocesseurs dédiés et d'outils puissants de développement.

Le concept de logique floue vient de la constatation que la variable booléenne, qui ne peut prendre que deux valeurs (vrai ou faux) est mal adaptée à la représentation de la plupart des phénomènes courants. Alors que la logique classique considère qu'une

proposition est soit vraie soit fausse, la logique floue distingue une infinité de niveaux de vérité (entre 0 et 1) [19].

Prenons l'exemple de la taille d'un homme : on veut déterminer si un homme est petit ou grand. En logique classique, on introduit une valeur "seuil": en deçà de cette valeur, l'homme sera petit et au-delà, il fera partie des grands.



Figure 3.13 Représentation de la logique classique

Mais cette discontinuité paraît absurde : à un demi centimètre près, l'homme peut passer d'une catégorie à l'autre. La notion de petite taille est difficilement modélisable avec la logique booléenne classique. En outre, en logique floue, on introduit des fonctions d'appartenance qui définissent, en fonction du paramètre (la taille), les degrés d'appartenance à chaque état figure (3.14).

Dans cet exemple, un homme de 1m70 sera grand à 50% et petit à 50%, et quelqu'un d'1m80 sera petit à 20% et grand à 80%.

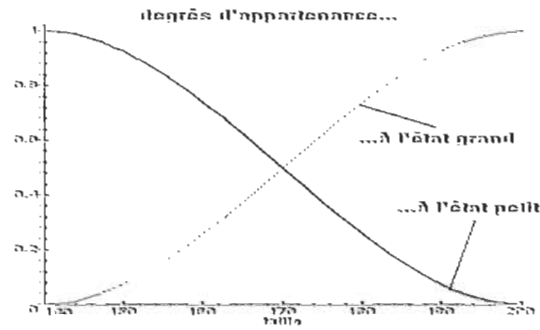


Figure 3.14 Représentation de la logique floue

Ces fonctions étant continues, des petites différences de taille font progressivement passer l'individu d'un état à l'autre, ce qui est beaucoup plus naturel.

Étapes du traitement.

- **Quantification floue (fuzzification).**

A partir des variables d'entrée et des fonctions d'appartenance, le système commence par déterminer le degré d'appartenance de chaque variable à chaque état. On constate qu'une variable peut être simultanément dans chaque état (un homme peut être un peu petit et un peu grand). On peut établir différentes courbes d'appartenance selon le problème (figures 3.15 à 3.17)

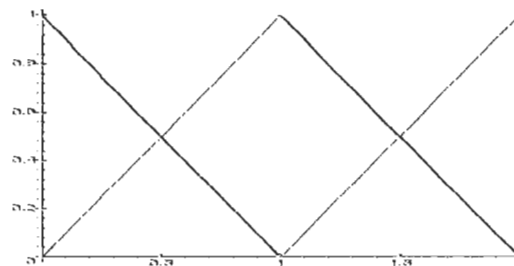


Figure 3.15 Fonctions d'appartenance en triangle

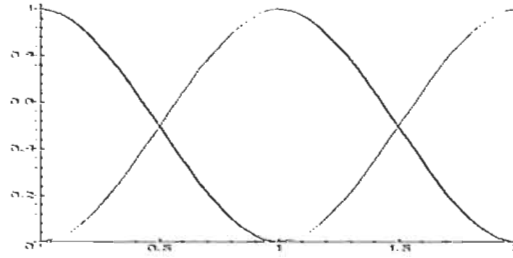


Figure 3.16 Fonctions d'appartenance en cloche

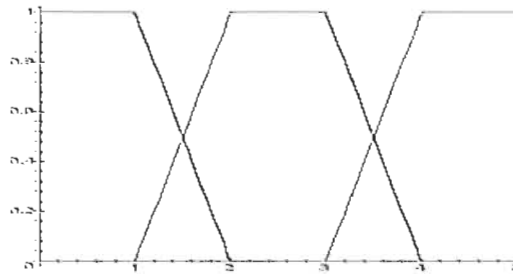


Figure 3.17 Fonctions d'appartenance en trapèze

Mais la forme des courbes est en général moins importante que leur nombre et leur placement (on utilise en général entre trois et sept états flous pour chaque variable).

- **Application des règles.**

Le contrôleur flou est basé sur un ensemble de règles du type si ... alors ... (comme en logique classique) établies en général de manière empirique.

Ex : Si la température est faible alors chauffer fort.

Ainsi, si la température est faible à 60%, la température de chauffe devra être forte à 60%. On peut évidemment, comme en logique classique, utiliser des connecteurs et, ou et non. Quand on utilise un ET, seul le degré d'appartenance minimum parmi les conditions est pris en compte. De même, pour un OU, on ne tient compte que du degré maximum. Le NON se traduit en retranchant la valeur de vérité de la variable à 1. (On peut

remarquer que ces formules restent vérifiées en logique booléenne). On peut aussi utiliser des connecteurs de précision du type environ, exactement, etc.... Leur définition précise peut varier, mais on peut par exemple, pour traduire un exactement, utiliser la fonction d'élévation au carré. La variable étant comprise entre 0 et 1, sa valeur va diminuer, sauf si elle est proche de 1. De même, pour un environ, on peut utiliser la fonction racine carrée.

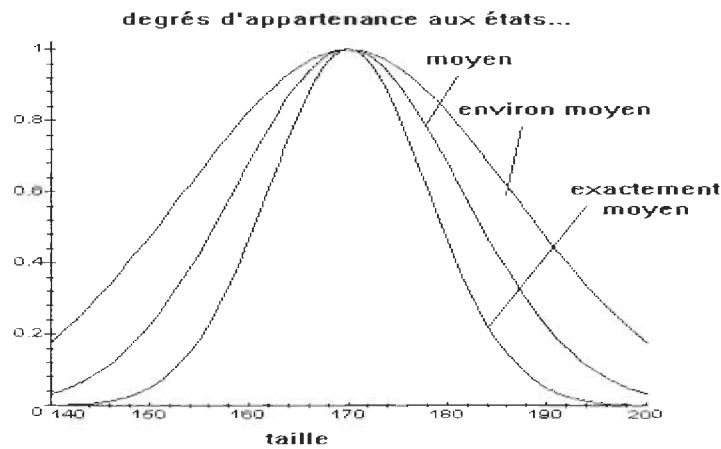


Figure 3.18 Connecteurs de précision

En appliquant les règles, on obtient le degré d'appartenance des variables de sortie à chaque état.

- **Défuzzification.**

Le problème de la *défuzzification* est d'obtenir une valeur précise à donner aux variables de sortie à partir des degrés d'appartenance.

Ex: Après application des règles, on obtient :

La puissance de chauffe doit être faible à 10%, moyenne à 20%, haute à 70% (voir figure 3.19).

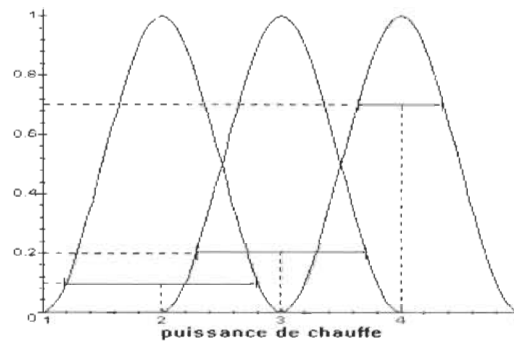


Figure 3.19 Résultat d'application des règles floues

Pour chaque état, on regarde la courbe au degré d'appartenance voulu. Les deux points d'intersection délimitent un segment. L'abscisse du milieu de ce segment détermine la valeur précise à utiliser pour cet état (dans l'exemple, la puissance de chauffe doit valoir 2 à 10%, 3 à 20% et 4 à 70%).

On obtient ainsi autant de valeurs qu'il y a d'états. A partir de ces chiffres, il existe plusieurs techniques pour savoir quelle valeur prendre en compte.

La technique du maximum est la plus simple. Elle consiste à ne prendre en compte qu'une valeur possédant le pourcentage maximum (la puissance de chauffe sera donc de 4). Cette technique est peu employée du fait de son imprécision : les valeurs secondaires apportent une certaine nuance dont on ne tient pas compte.

La technique de la moyenne pondérée est plus évoluée : on prend comme valeur de sortie la moyenne de ces différentes valeurs pondérées par leur pourcentage.

Dans notre exemple, la puissance de chauffe sera de

$$\frac{2 \times 0,1 + 3 \times 0,2 + 4 \times 0,7}{0,1 + 0,2 + 0,7} = 3,6$$

Cette méthode, également simple à mettre en œuvre s'avère assez efficace.

La technique du centre de gravité consiste à tracer les surfaces délimitées par chaque courbe en dessous du degré d'appartenance voulu (figure 3.20).

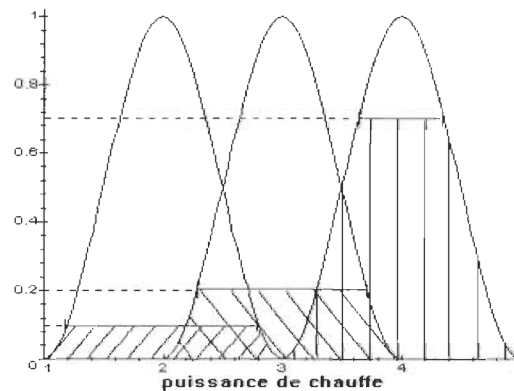


Figure 3.20 Technique de centre de gravité

L'abscisse du centre de gravité des surfaces obtenues fournit la valeur de sortie. Cette méthode, bien que plus complexe, donne les meilleurs résultats et est très couramment employée.

Avantages et inconvénients.

La logique floue s'avère très simple à mettre en œuvre par rapport aux techniques traditionnelles de contrôle basées sur des modèles mathématiques. Ce genre de système utilise un ensemble d'équations différentielles qui permettent de calculer les réponses du système à partir des entrées. Ces systèmes s'avèrent très efficaces, mais nécessitent de

longues années de développement et d'étude théorique. De plus, il n'existe pas toujours de modèle mathématique pour toutes les situations, et s'il existe, ce modèle peut être très coûteux en temps de calcul ou en espace mémoire. Ainsi, un système basé sur des règles empiriques peut être plus efficace. Mais le caractère empirique de ces règles est par ailleurs un inconvénient, parce que ces règles ne sont pas précises, et peuvent être sources d'erreurs. Il faut donc passer par une phase de test du contrôleur afin de corriger ces règles.

Domaines d'application.

Généralement, la logique floue s'avère très efficace dans les systèmes pour lesquels il n'existe pas de modélisation, pour les systèmes contrôlés par des humains et dans tous les domaines où un "flou" persiste, comme l'économie, les sciences naturelles et les sciences humaines. C'est pourquoi la logique floue est utilisée dans le domaine juridique pour simuler des procès, pour la prévision des tremblements de terre, en médecine pour diagnostiquer le cancer, en économie pour analyser le marché et en informatique, pour la reconnaissance de caractères et la commande vocale.

La logique floue va être employée dans la structure du contrôleur en MG pour faire l'approximation linéaire de la fonction discontinue « signe ». Cette approximation est essentielle pour minimiser les oscillations en adoucissant les transitions autour de la ligne de commutation.

3.3 Conclusion

Éliminer les effets de friction et flexion dans les manipulateurs robotiques actuels devient un véritable défi. Pour cela, les aspects de la conception des systèmes, comprenant les concepts matériels tels que la lubrification et aussi les possibilités de contrôle, doivent être considérés.

Aujourd'hui, les phénomènes de friction et flexion présents dans les systèmes automatisés sont mieux compris et des modèles de plus en plus précis de ces phénomènes comprenant notamment la dynamique de friction et de flexion sont disponibles.

Ainsi, de nombreuses techniques de compensation ont été développées, méthodes généralement développées empiriquement puisqu'il est difficile d'obtenir des modèles exacts pour ces phénomènes.

Les méthodes modernes de commande utilisant les réseaux de neurones et la logique floue prennent plus de place et résolvent des problèmes associés à la commande des systèmes non linéaires.

La méthodologie adoptée dans cette étude, pour commander une articulation flexible avec friction incorpore les techniques discutées dans ce chapitre et va être détaillée dans le chapitre suivant.

Chapitre 4

Commande de position d'une articulation flexible avec friction

La qualité de la commande des systèmes électromécaniques dépend pour une part importante de la précision du modèle à partir duquel les forces ou les couples sont calculés. Cette notion de qualité de la commande est délicate à formuler d'une manière générale, indépendamment d'une application particulière.

La méthodologie de synthèse de lois de commande vise à améliorer les performances en abandonnant le découplage fréquentiel simplificateur et en faisant intervenir explicitement la dynamique, c'est-à-dire la relation entre forces et accélérations. Une première étape dans ce sens consiste donc à proposer une commande de base fondée d'une façon ou d'une autre sur les résultats théoriques appliqués à une double intégration: les améliorations ont ensuite pour but de raffiner cette commande de base par l'adjonction de sophistications variées: découplage linéaire ou non-linéaire, suppression des vibrations, compensations diverses, que l'on peut regrouper sous la terminologie usuelle de "commande dynamique" [20].

Le modèle d'une articulation flexible avec friction n'est pas exact et les méthodes classiques pour dériver le modèle inverse ne peuvent être utilisées (à cause de la discontinuité présente dans le modèle) [21]. Les RNA sont utilisés avec succès dans la conception des systèmes de commande industrielle [22]. Plusieurs méthodes, pour construire et faire apprendre le réseau, sont utilisées. Comme les RNA sont capables de faire l'approximation d'une fonction non-linéaire quelconque (avec la bonne structure et les bons poids), nous allons exploiter l'utilisation des RNA pour la modélisation inverse approximative d'un joint robotique flexible et nous l'utiliserons dans la boucle de commande de position.

4.1 Commande inverse des articulations flexibles avec friction

4.1.1 Modélisation inverse avec RNA

Le modèle dynamique inverse des articulations flexibles va être appris par un RNA. Après l'apprentissage du réseau, il va être utilisé pour commander le système pour suivre la trajectoire désirée par le signal d'anticipation (le couple électromagnétique). La méthode pour construire et faire apprendre le réseau est montrée dans [22] où on montre l'application d'un RNA d'anticipation (*feedforward*) pour l'identification du modèle inverse et la commande des systèmes dynamiques.

Dans cette section nous présentons l'approche de calcul pour l'apprentissage du modèle dynamique inverse d'un joint robotique flexible et son utilisation dans la méthode de commande appelée ainsi commande inverse. La figure 4.1 montre cette approche. Dans cette figure, le bloc nommé « RNA » est une boîte noire avec ses entrées et sorties. Cette boîte noire n'est que le réseau de neurones artificiels. Cette approche est appelée l'apprentissage général (hors ligne) [23]. Dans cette méthode, la construction du modèle inverse est faite avant de l'utiliser dans l'algorithme de commande.

La sortie du processus sous contrôle (y) correspond à l'entrée connue (u) et est utilisée comme entrée du RNA pour obtenir sa sortie (u_c). Le processus d'apprentissage prend lieu en minimisant l'erreur de position instantanée (e^2) entre (u) et (u_c). Après la phase d'apprentissage, les poids des connections entre neurones du réseau sont fixés, puis le RNA est utilisé dans le contrôleur devant le processus. Le succès avec

cette méthode dépend largement de la capacité du RNA de généraliser ou d'apprendre à répondre correctement aux entrées qui n'ont pas été utilisées dans la phase d'apprentissage.

La méthode générale d'apprentissage (hors ligne) est utilisée dans cette étude pour développer l'algorithme de commande. La méthode d'apprentissage spécialisée (en ligne) doit être utilisée pour faire face au problème de la grande quantité de données d'apprentissage nécessaire, car normalement on ne connaît pas les entrées essentielles et désirées du processus sous contrôle (ce qui pourrait être le sujet d'une étude future).

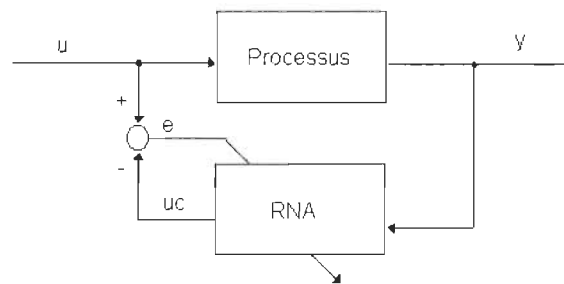


Figure 4.1 Architecture d'apprentissage général.

L'articulation reçoit le couple $T(t)$ comme entrée et on trouve la trajectoire $\theta(t)$ à la sortie. Le modèle dynamique inverse est mis au sens opposé d'entrée-sortie de celui de l'articulation. Il reçoit la trajectoire comme entrée et il génère le couple comme sortie (Figure 4.1).

Les données d'apprentissage sont fournies à partir des données de simulation qui représentent les paramètres mesurés de l'articulation flexible (c.-à-d. position, vitesse et accélération de la charge et du moteur) obtenues avec le modèle de l'articulation en réalisant des trajectoires prédéfinies (figure 4.2). Ces trajectoires ont des

caractéristiques qui font en sorte d'exciter tous les modes internes du modèle et qui font ressortir les effets de friction sur le moteur et la charge.

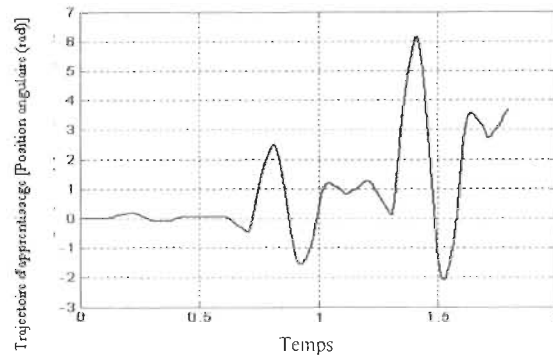


Figure 4.2 Trajectoire d'apprentissage.

Après une série d'évaluations sur la performance de suivi de trajectoire des différentes structures du RNA multicouche d'anticipation (*feed-forward*), on a choisi un réseau qui a donné un meilleur compromis entre la complexité de la structure (c.-à-d. le nombre de couches et le nombre de neurones dans chaque couche) et la performance d'apprentissage. Ce réseau est composé de 6 neurones dans la couche d'entrée, 10 dans la couche cachée et 1 à la sortie. Il a été créé dans l'environnement de simulation Simulink® de Matlab® avec la fonction *Newff*, qui génère un réseau avec des valeurs aléatoires de poids.

Les fonctions d'activation dans les neurones utilisés dans la première couche sont des fonctions non linéaires de type linéaire saturé (Figure 4.3).

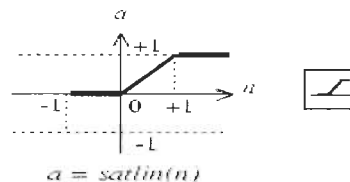


Figure 4.3 Fonction linéaire saturée.

Les fonctions d'activation dans les neurones utilisés dans la couche cachée sont les fonctions non linéaires de type sigmoïde (Figure 4.4).

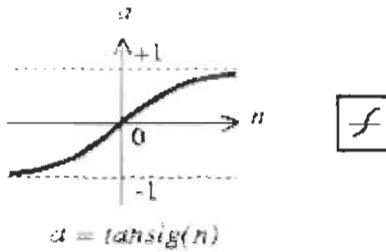


Figure 4.4 Fonction Sigmoïde.

Le fonction d'activation dans le neurone utilisé dans la couche de sortie est la fonction linéaire purelin (Figure 4.5).

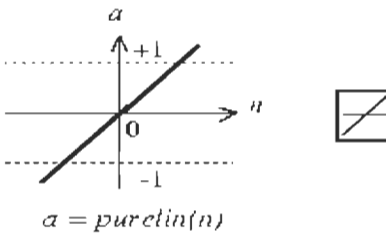


Figure 4.5 Fonction purelin.

Pour l'apprentissage du réseau, une version de la méthode de rétro-propagation (*Back propagation*) appelée (*Levenberg-Marquardt Back propagation*) est utilisée[24]. Cette méthode est implantée dans Matlab® par la fonction *trainlm* qui peut entraîner tous les réseaux si leurs poids, entrées, et fonctions de transfert sont dérivables.

La rétro-propagation est utilisée pour calculer le Jacobien « jX » d'un indice de performance « $perf$ » par rapport au vecteur des poids et biais du réseau « X ». Ce vecteur est ajusté selon la méthode de Levenberg-Marquardt,

$$jj = jX * jX$$

$$je = jX * E$$

$$dX = -(jj + I * \eta) \setminus je$$

où « E » représente le vecteur d'erreur, « I » est la matrice d'identité.

La valeur adaptative « η » est augmentée par « $\Delta^+ \eta$ » jusqu'à ce qu'une dégradation de la valeur de l'indice de performance « $perf$ » soit notée. On change les paramètres du réseau « X » et on baisse la valeur de « η » par « $\Delta^- \eta$ ».

L'entraînement s'arrête quand une de ces conditions se produit :

1. Le nombre maximum d'époques (répétitions) est atteint.
2. Le temps maximal a été dépassé.
3. L'indice de performance atteint son but «*goal*».
4. Le gradient de l'indice de performance tombe au-dessous d'une valeur minimale fixe «*mingrad*».
5. La valeur de η dépasse une valeur maximale « η_{max} »
6. La performance de validation a été «*max_fail*» fois augmentée depuis la dernière diminution (quand validation est utilisée).

Les paramètres (les valeurs en *italique*) de cette méthode, ainsi que l'architecture du réseau, peuvent être programmés.

Les résultats de simulation utilisant le modèle inverse avec RNA avec apprentissage hors ligne et sans boucle de commande sont montrés au Chapitre 5. Ces résultats montrent que le RNA est capable d'apprendre le modèle inverse mais, la performance n'est pas satisfaisante et nous avons besoin d'améliorations qui ont ensuite pour but de raffiner cette commande de base par l'adjonction de sophistications variées.

4.2 Algorithme de commande inverse

4.2.1 Développements théoriques.

L'algorithme de commande proposé peut être résumé comme suit : Un RNA qui a appris le modèle inverse du système électromécanique, c.-à-d. d'une articulation flexible avec friction, fournit le signal d'anticipation nécessaire (le couple électromagnétique) pour appliquer à l'entrée du système et ce à partir de la trajectoire de consigne, que le système doit suivre. La trajectoire de consigne est un vecteur qui se compose de la position, la vitesse et l'accélération désirées de la charge à chaque instant. Cette trajectoire est présentée à l'entrée de RNA qui génère le signal d'anticipation (c.-à-d. le couple électromagnétique).

Pour améliorer la performance du système, un signal de rétroaction est généré à partir du vecteur des erreurs du système. Ce signal vient d'un contrôleur par mode de glissement (MG) avec une bande d'adoucissement adaptative réalisée par la logique floue (la fonction discontinue « Signe » est réalisée par la logique floue pour minimiser les oscillations). Le diagramme bloc de cette structure est montré à la figure 4.6.

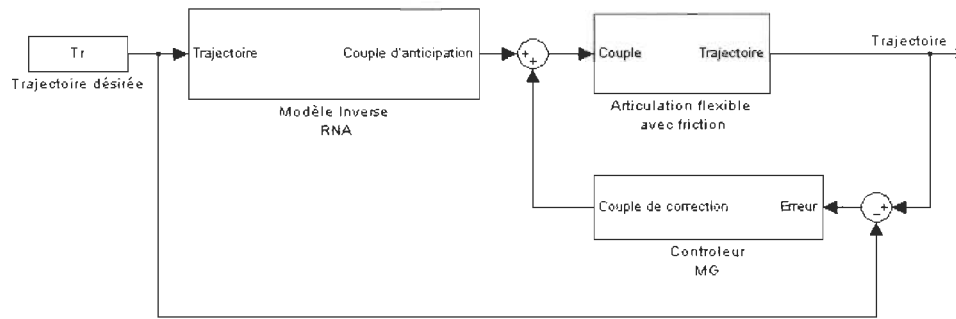


Figure 4.6 Diagramme bloc du système

Le modèle Simulink du diagramme bloc pour développer et évaluer l'algorithme de commande proposé est montré à la figure 4.7.

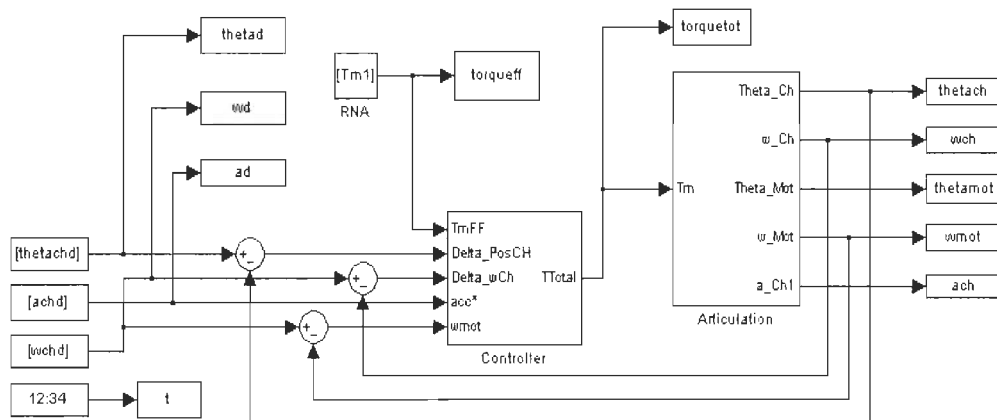


Figure 4.7 Modèle Simulink® du système

Les étapes de conception du correcteur fonctionnant en MG sont :

1. Choisir la surface de commutation.
2. Déterminer une loi de commande assurant l'existence du mode de glissement et la convergence vers la surface de commutation.

3. Éliminer les oscillations sur le signal de commande lorsque celles-ci sont indésirables.

Analysons donc chacune des ces étapes et la théorie s'y rapportant.

1. Choix de la surface de commutation :

La surface de commutation définit le comportement du système en mode de glissement, et ainsi le comportement du système en boucle fermée si le mode de glissement est maintenu.

Pour déterminer l'équation de la surface de glissement, plusieurs méthodes peuvent être employées dont le positionnement des pôles et la minimisation d'un critère quadratique (commande optimale) [3]. Les pôles obtenus doivent définir un système stable (partie réelle négative). De plus, un modèle d'ordre réduit étant utilisé pour représenter le système, les pôles doivent être choisis de façon à ce que les modes du système non modélisés à l'étape de conception ne soient pas excités. Ceci implique une rapidité spécifiée limite du système (limite de la partie réelle des pôles). Cette limitation de la rapidité spécifiée du système peut aussi être importante afin de minimiser le temps de convergence vers la surface de glissement suite à une condition initiale se trouvant hors de cette surface. En effet, plus les pôles dominants spécifiés du système sont rapides, plus la phase de convergence peut être longue suite à une variation de consigne (Figure 4.8).

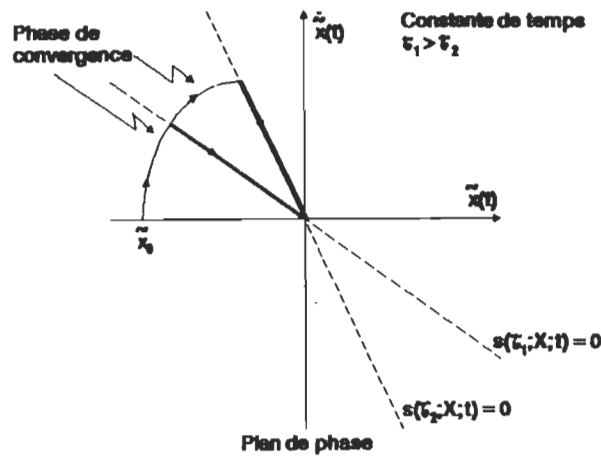


Figure 4.8 Effet de la variation de la constante de temps spécifiée sur le temps de convergence vers la droite de glissement suite à un échelon d'erreur.

Il est très important de minimiser la durée de cette phase car le système est sensible aux variations de paramètres et aux perturbations pendant cette phase. Selon ces critères, les pôles pourraient être positionnés dans une région telle que définie par la zone hachurée de la figure 4.9.

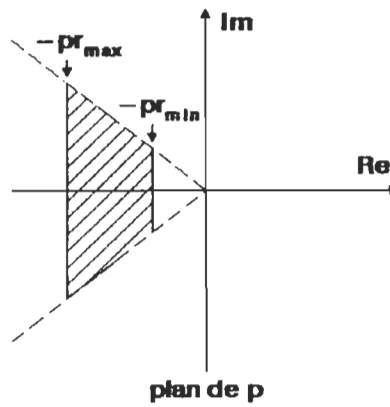


Figure 4.9 Domaine des pôles imposés.

Notons que pour assurer la convergence vers la surface de glissement et diminuer le temps de convergence, un gain élevé peut être nécessaire, et ce, en particulier si la surface de glissement définit un système très rapide. Ainsi, suite au choix d'une

surface de glissement et d'une loi de commande, les signaux doivent être analysés afin de voir si les modes non modélisés du système ne sont pas excités par la présence des gains trop élevés. Le processus de conception devient alors itératif.

L'ordre minimum pouvant être obtenu pour le système correspond à la différence entre l'ordre du système original et l'ordre de la loi de commande [3]. Dans le cas de notre système de positionnement, l'ordre du système étant faible, nous tenterons d'obtenir une caractéristique du premier ordre. Un seul pôle λ réel et négatif sera donc employé selon

$$-\text{Pr}_{\max} < \lambda < -\text{Pr}_{\min}$$

où Pr_{\min} définit la rapidité minimale acceptable et Pr_{\max} la limite en fréquence pour ne pas exciter les modes non modélisés. Nous obtenons ainsi une surface $S(X;t)$ variant dans le temps qui représente une droite de glissement (Figure 4.10) sur laquelle $s(X;t) = 0$, $s(X;t)$ étant défini par

$$s(X;t) = \ddot{\tilde{x}}(t) + \lambda \bullet \tilde{x}(t)$$

où $\tilde{x}(t) = x_d(t) - x(t)$, $X(t)$ est le vecteur des variables d'état $[x(t) \ \dot{x}(t) \dots \ x^{(n-1)}(t)]^T$ ($x^{(m)}$ représente la dérivée $m^{\text{ième}}$ de x), $X_d(t)$ est le vecteur des valeurs de référence $[x_d(t) \ \dot{x}_d(t) \dots \ x_d^{(n-1)}(t)]$. $s(X;t)$ représente alors l'erreur du point représentatif du système à la trajectoire désirée.

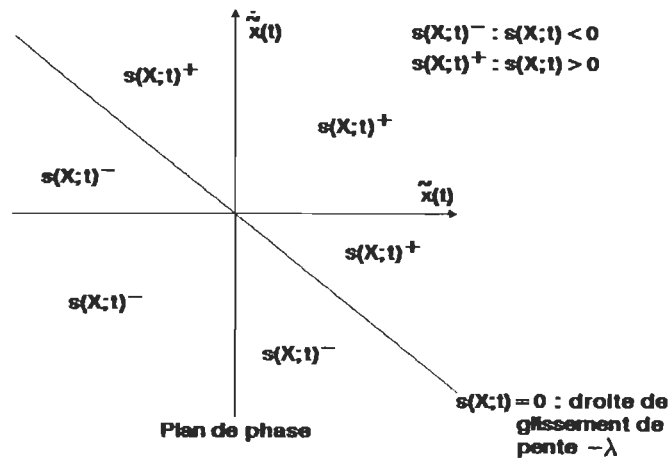


Figure 4.10 Définition du plan de phase pour un système avec caractéristique spécifiée du premier ordre.

Pour l'articulation flexible le vecteur d'états X est composée de $[\theta \quad \dot{\theta} \quad \ddot{\theta}]$ qui sont la position, la vitesse et l'accélération de la charge respectivement et qui définissent la trajectoire réalisée. X_d , le vecteur d'état désiré, est aussi la trajectoire désirée de la charge.

2. Développement de la loi de commande :

Puisqu'on est en présence d'un système pouvant être représenté par une équation différentielle continue par partie avec terme discontinu, la loi de commande n'est pas unique. La forme de loi de commande la plus générale est constituée d'une contre-réaction d'état et d'un terme discontinu permettant de compenser l'effet des variations de paramètres et des perturbations. De plus, plusieurs méthodes peuvent être utilisées afin de déterminer la loi de commande. Voyons une méthode simple utilisant directement les conditions de glissement et de convergence et qui permet d'obtenir une loi de commande avec retour d'état et terme de compensation discontinu [3].

La procédure de conception de la loi de commande est la suivante

1. Modéliser le système.
2. Exprimer $\dot{s}(X;t)$ en fonction de $u(X;t)$.
3. Déterminer les différents termes de la loi de commande permettant de satisfaire les conditions de glissement et de convergence qui sont :
 - 3.1 Déterminer $u_1(X;t)$ permettant de compenser tous les termes connus.
 - 3.2 Déterminer $u_2(X;t)$ permettant de compenser les termes incertains et les perturbations. Ce terme sera discontinu. $u_1(X;t)$ et $u_2(X;t)$ permettent d'assurer le mode de glissement.
 - 3.3 Ajouter un terme $u_3(X;t)$ et $u_4(X;t)$ permettant d'assurer la convergence et améliorant la performance.
 - 3.4 La loi de commande $u(X;t)$ est obtenue en additionnant $u_1(X;t)$, $u_2(X;t)$, $u_3(X;t)$ et $u_4(X;t)$.

Reprenons le développement de la loi de commande d'ordre n [3] pour un système d'ordre 2 (en négligeant les termes d'ordre élevé) dont la grandeur de sortie et les dérivées première et seconde de la grandeur de sortie sont accessibles.

Notre système est représenté sous forme canonique par :

$$\ddot{x}(t) = f(X;t) + b(X;t) \bullet u(X;t) + d(t)$$

où $f(X;t)$ représente le modèle dynamique, $u(X;t)$ est le signal de commande, X est le vecteur d'état, $d(t)$ représente les perturbations et $b(X;t)$ est le gain d'entrée (positif et éventuellement variable).

Définissant $\hat{f}(X;t)$ et $\hat{b}(X;t)$ comme valeurs nominales de $f(X;t)$ et $b(X;t)$ et en substituant dans les équations de l'algorithme [3]

$$f(X;t) = \hat{f}(X;t) + \Delta f(X;t)$$

$$\frac{1}{\hat{b}} = 0.5 * \left\{ \left[\frac{1}{b(X;t)_{\min}} \right] + \left[\frac{1}{b(X;t)_{\max}} \right] \right\}$$

$$\Delta \left(\frac{1}{\hat{b}} \right)_{\max} = 0.5 * \left\{ \left[\frac{1}{b(X;t)_{\min}} \right] - \left[\frac{1}{b(X;t)_{\max}} \right] \right\}$$

Définissons la réponse désirée par la loi de commutation suivante qui décrit un système de premier ordre avec bande passante de λ :

$$\dot{s}(X;t) = \ddot{\tilde{x}}(t) + \lambda \dot{\tilde{x}}(t) \quad \text{où } \tilde{x}(t) = x_d(t) - x(t)$$

Nous obtenons ainsi

$$\dot{s}(X;t) = \ddot{\tilde{x}}(t) + \lambda \dot{\tilde{x}}(t)$$

Exprimons $\dot{s}(X;t)$ en fonction de $u(X;t)$:

$$\dot{s}(X;t) = -\hat{f}(X;t) - b(X;t) * u(X;t) - d(t) + \ddot{x}_d(t) + \lambda \dot{\tilde{x}}(t)$$

On obtient ainsi la loi de commande qui satisfait la condition de glissement, compense tous les termes connus, compense les termes incertains et les perturbations et qui assure la convergence [3]

$$u(X;t) = \frac{-\hat{f}(X;t) + \ddot{x}_d(t) + \lambda \dot{\tilde{x}}(t)}{\hat{b}} + \left[\frac{\Delta \left(\frac{1}{\hat{b}} \right)_{\max} * \left| -\hat{f}(X;t) + \ddot{x}_d(t) + \lambda \dot{\tilde{x}}(t) \right|}{+ \frac{|\Delta f(X;t)| + D}{b(X;t)_{\min}} + k * |s(X;t)|} \right] * \text{sign}(s(X;t))$$

où D représente l'amplitude maximale de $d(t)$, $k \geq \frac{\beta}{b(X;t)_{\min} * |s(X;t)|}$ est un gain pour

assurer la convergence et β est une fonction de classe K (continue et non décroissante). Si cette condition est satisfaite, alors, suite à toute condition initiale hors de $S(X;t)$, les trajectoires se dirigeront vers $S(X;t)$ [3].

Ainsi, considérant l'équation du système à commander (la dynamique du moteur et les termes de friction sont assimilés à une perturbation):

$$\ddot{\theta}_L(t) = \frac{-J_M(X;t)\ddot{\theta}_M(t)}{J_L(X;t)} + \frac{\tau_M(t)}{J_L(X;t)} - \frac{[\tau_{FM}(\dot{\theta}_M) + \tau_{FL}(\dot{\theta}_L)]}{J_L(X;t)}$$

$$\ddot{\theta}_L(t) = \frac{-\tau_{FL}(\dot{\theta}_L)}{J_L(X;t)} + \frac{\tau_M(t)}{J_L(X;t)} - \frac{J_M(X;t)\ddot{\theta}_M(t) - \tau_{FM}(\dot{\theta}_M)}{J_L(X;t)} = f(X;t) + b(X;t)u(t) + d(X;t)$$

Alors :

$$b = \frac{1}{J_L(X;t)}$$

$$u_1 = u_{ff} + (\ddot{x}_d(t) + \lambda \tilde{x}) J_L(X;t)$$

où u_{ff} est le signal d'anticipation fourni par le RNA d'anticipation.

$$u_2 = \left[\Delta \left(\frac{1}{b} \right)_{\max} |\hat{f}| + \frac{|\Delta f| + D}{(b)_{\min}} \right] * \text{sign}(s) + \Delta \left(\frac{1}{b} \right)_{\max} |\ddot{x}_d + \lambda \tilde{x}| * \text{sign}(s)$$

où D doit compenser, en plus des imprécisions sur les paramètres, les erreurs sur le signal d'anticipation u_{ff} . Le terme u_2 est approximé par le terme suivant pour simplifier l'implantation de la loi de commande.

$$u_2 = K_0 \text{sign}(S)$$

$$u_3 = k |s(X;t)| * \text{sign}(s) = k * s + k1 * \dot{S}$$

où le terme en \dot{S} a été ajouté pour améliorer la stabilité autour de la droite de glissement (le correcteur est de type proportionnel intégral).

3. Commande par mode de glissement avec bande d'adoucissement

L'activité intense du signal de commande constitue un des problèmes principaux de la commande par mode de glissement. Ce phénomène est causé par les discontinuités du signal de commande sur la surface de glissement et est amplifié par l'effet des faibles

délais et des modes du système non modélisés lors de la conception. La forte activité du signal de commande peut causer une détérioration rapide des éléments mécaniques du système, provoquer des oscillations sur le signal de sortie et causer une erreur à la sortie en régime permanent.

Comme nous l'avons mentionné, nous allons utiliser la méthode de substitution du terme discontinu de la loi de commande par une fonction d'adoucissement qui est définie par une bande de largeur variable avoisinant l'hyperplan de glissement à l'intérieur de laquelle le gain est variable. L'adoucissement du signal de commande est réalisé en substituant la fonction discontinue $Sign(s)$ par une approximation de fonction $Sign(s)$ réalisée par la logique floue. La fonction résultante est implantée par une table de relations.

La figure 4.11 montre la structure du contrôleur dans l'environnement Simulink. Un bloc de délai « $1/z$ » avec un bloc de bande morte est utilisé pour obtenir l'approximation S_d de la dérivée de S .

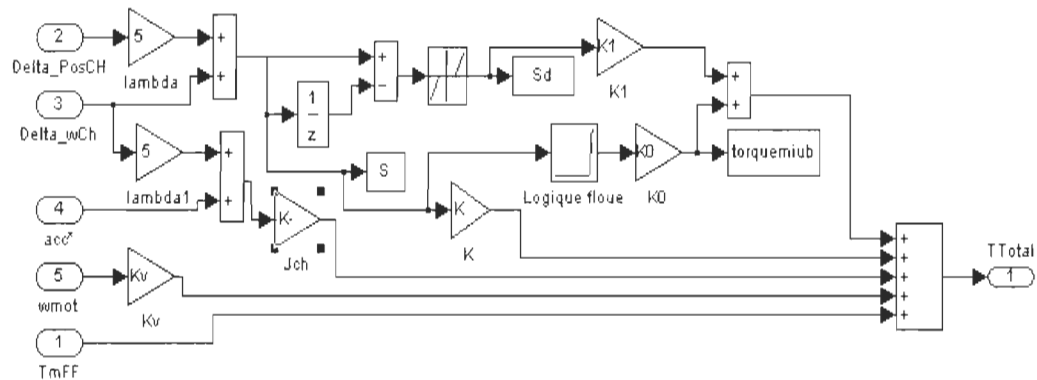


Figure 4.11 Structure de contrôleur en MG.

Pour la stabilisation interne du système et pour éliminer les oscillations côté moteur, un terme en vitesse du moteur est introduit avec un gain de K_v . Les gains K , $K_1 = J_{ch}$, K_0 et K_v sont choisis pour avoir la meilleure performance (c.-à-d. erreur minimum de position de la charge.) après une série de simulations en variant, une à la fois, ces valeurs. Les résultats de simulation sont montrés dans le chapitre suivant.

4.3 Conclusion

Dans ce chapitre, le développement théorique de la méthodologie proposée pour la commande d'une articulation flexible avec friction est présenté.

Cette méthode est basée sur l'identification du modèle inverse du système avec un RNA. À partir du modèle inverse et de la trajectoire désirée, le signal d'anticipation est généré. Ce signal représente le couple électromagnétique nécessaire pour que l'articulation réalise la trajectoire désirée.

Pour compenser l'inexactitude du modèle inverse, désensibiliser le système contre les perturbations internes et externes et augmenter la performance du système, un contrôleur par mode de glissement est introduit dans la boucle de rétroaction. Les détails du développement et de l'implantation de ce contrôleur sont présentés, ainsi que l'utilisation de la logique floue pour appliquer la bande d'adoucissement qui sert à éliminer les oscillations internes.

Le chapitre suivant va présenter les résultats de simulation et mettre en évidence les avantages et les limites de cette méthode.

Chapitre 5

Résultats de simulation

et analyse de performances

5.1 Performances dynamiques et statiques de la loi de commande

La performance de la loi de commande proposée est montrée par des simulations dans l'environnement de développement Matlab®. Les résultats de simulations sous différentes conditions sont présentées et sont analysées. Les paramètres de l'articulation simulée sont [13]:

L'inertie de la charge : $J_L = 5.5 \times 10^{-5} \text{ N}\cdot\text{m}/(\text{rad}/\text{s}^2)$;
L'inertie du moteur : $J_M = 5.5 \times 10^{-5} \text{ N}\cdot\text{m}/(\text{rad}/\text{s}^2)$;
Ratio de boîte de vitesse : $N = 1$;
Ressort de torsion : $K = 0.52 \text{ N}\cdot\text{m}/\text{rad}$;
Période d'échantillonnage : $T_s = 0.0001 \text{ s}$;
Efficacité de transmission : $\eta = 1$ (>95% approximé à 1 pour ne pas introduire des termes non linéaires supplémentaires);
Coefficients de frottements de Coulomb, visqueux et statique, et la vitesse caractéristique de frottement statique : $F_c = 0.0288 \text{ Nm}$, $F_v = 0.00055 \text{ Nm}/(\text{rad}/\text{s})$, $F_s = 0.0191 \text{ Nm}$, $v_s = 0.95 \text{ rad/s}$

Le signal de consigne, position et vitesse de la charge, utilisé dans cette étude est montrée dans la figure 5.1.

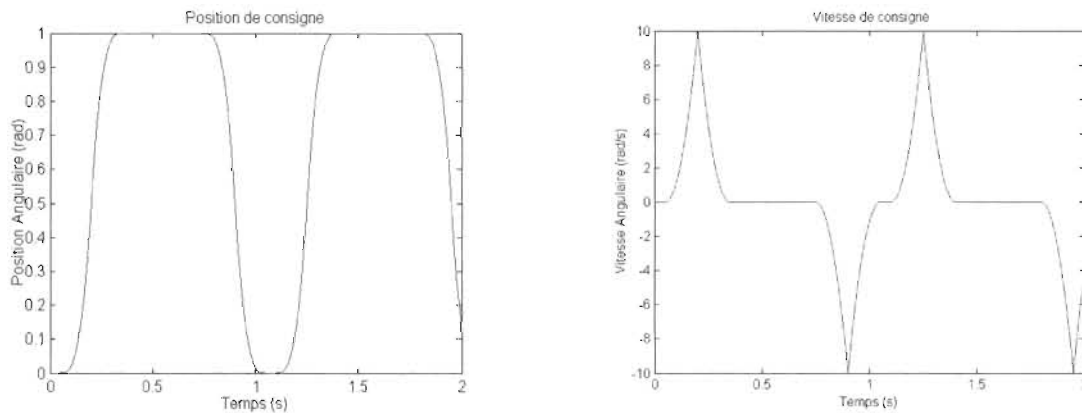


Figure 5.1 Position et vitesse de consigne.

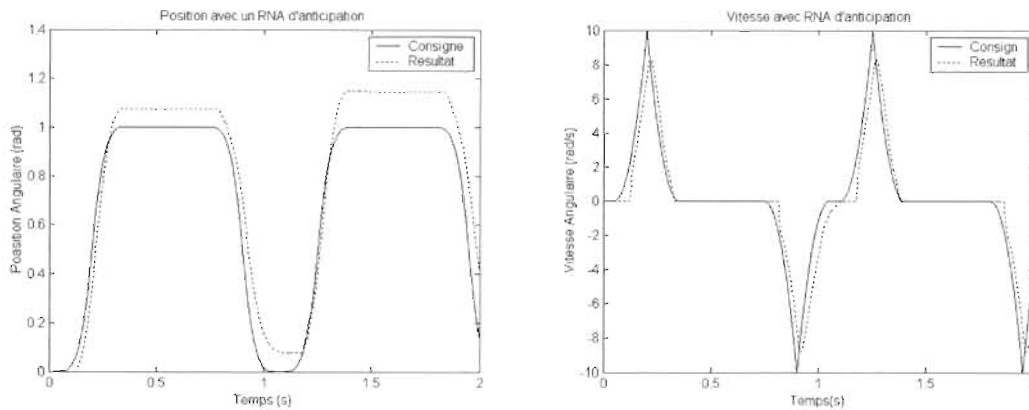
Trois types de simulation seront utilisés pour la validation de la loi de commande. Une avec le réseau de neurones seul sans le contrôleur, une avec le réseau de neurones et le contrôleur par mode glissement, et la dernière avec le réseau de neurones et le contrôleur

par mode de glissement en introduisant un terme de vitesse de moteur pour la stabilisation interne. Nous allons considérer les deux cas de conditions initiales nulles et non nulles. Dans le cas des conditions non nulles la position initiale de la charge est de 0.25 rad. La performance du système va être mesurée en utilisant deux fonctions de l'erreur de position « e » : la première $= \sqrt{\sum e^2}$ et la deuxième $= \text{Max}|e|$.

5.1.1 Réponse à la consigne sans le contrôleur

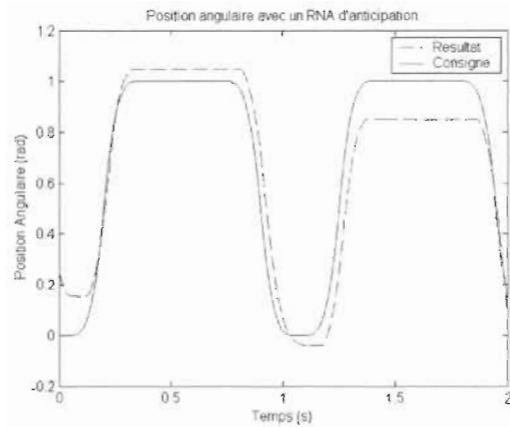
Une simulation est réalisée utilisant le RNA qui génère le couple électromagnétique d'anticipation et montre la capacité du réseau à apprendre le modèle inverse de l'articulation flexible avec friction.

La figures 5.2 montre les résultats de simulation avec conditions initiales nulles (a, b) et non nulles (c, d). On constate que le RNA a appris le dynamique du système mais qu'il y a une erreur stationnaire.

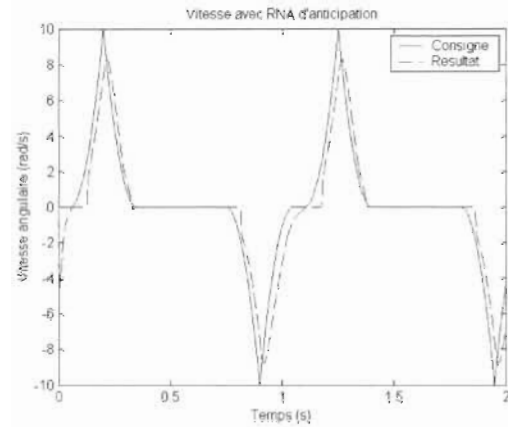


a) Position avec conditions initiales nulles b) Vitesse avec conditions initiales nulles

Figure 5.2 Résultats avec RNA d'anticipation.



c) Position avec conditions initiales non nulles



d) Vitesse avec conditions initiales non nulles

Figure 5.2 (suite).

La figure 5.3 montre le comportement de la vitesse du moteur avec conditions initiales nulles. Nous pouvons observer que le moteur inverse sa vitesse à la fin des mouvements afin de freiner la charge, en fait, afin de dissiper l'énergie accumulée dans le ressort de torsion. Nous notons aussi la très bonne stabilité interne du système.

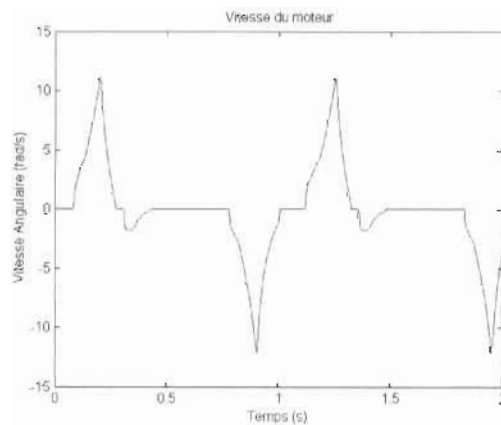


Figure 5.3 Vitesse du moteur avec RNA d'anticipation.

La figure 5.4 montre le couple d'anticipation qui est appliqué à l'entrée de l'articulation. Ce couple est généré par le RNA (le modèle inverse).

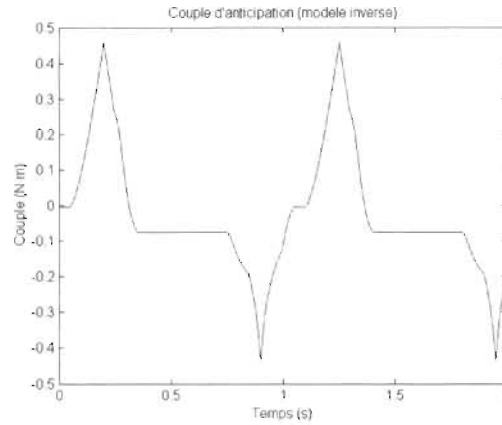


Figure 5.4 Couple d'anticipation à la sortie du RNA.

Les indices de performance pour le système avec RNA d'anticipation sans le contrôleur sont présentés dans le tableau 5.1. Dans la figure 5.5 on constate une valeur relativement importante de l'erreur initiale et de l'erreur résiduelle de position (à vitesse nulle) ainsi qu'une dérive graduelle de l'erreur et ce à cause des imprécisions du modèle. Ces résultats montrent une faible performance et la nécessité d'introduire un contrôleur pour l'augmenter.

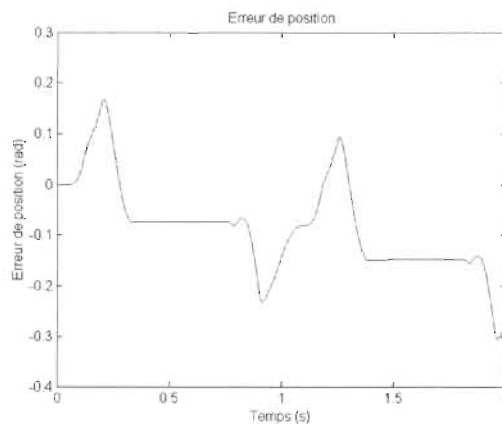


Figure 5.5 Erreur de position avec RNA d'anticipation.

5.1.2 Réponse à la consigne avec contrôleur MG

Les résultats montrés ci-après sont obtenus avec le RNA et le contrôleur MG développés dans le chapitre précédent (Figure 4.11). Les valeurs des gains utilisés dans le contrôleur sont obtenues suite à une série de simulations (voir l'Annexe C pour plus des détails). On a trouvé que les valeurs des gains qui donnent l'indice de performance minimal sont : $K = 0.6$, $K_0 = 22$, $K_1 = 3.5$. La valeur de $\lambda = 5$ dans l'équation du contrôleur est de l'ordre de la fréquence naturelle du système.

La figure 5.6 montre les résultats de simulation. On constate l'amélioration de performance de suivi de consigne de position et de vitesse mais avec un peu d'oscillations dans les résultats de la vitesse.

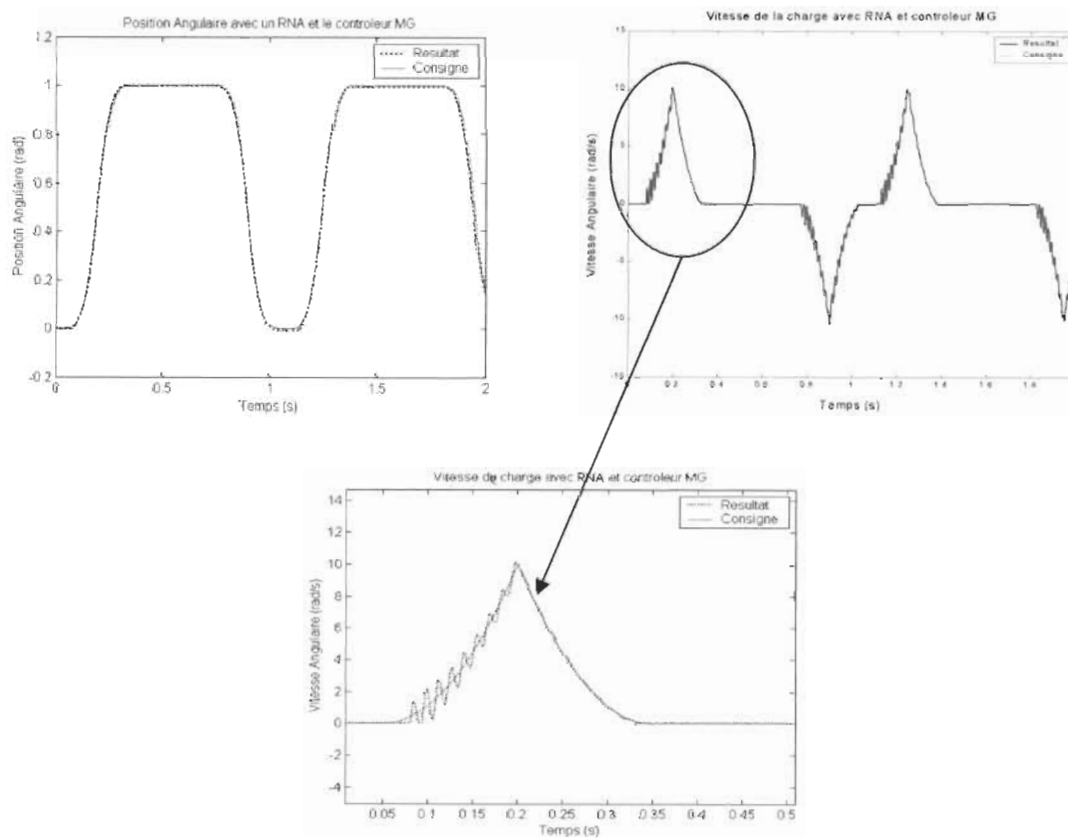


Figure 5.6 Résultats avec RNA d'anticipation et contrôleur MG.

La figure 5.7 montre les oscillations dans le couple de commande qui causent les oscillations de vitesse. Notons que le système est stable (les oscillations convergent vers 0).

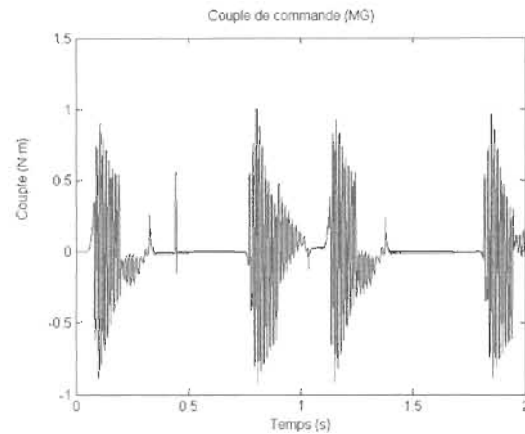


Figure 5.7 Couple de commande à la sortie du contrôleur MG.

La figure 5.8 montre la grande performance du contrôleur où l'erreur de position est très faible avec peu d'oscillations.

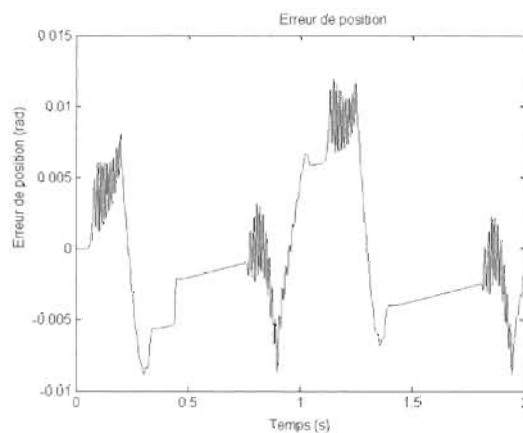


Figure 5.8 Erreur de position avec contrôleur MG.

Avec des conditions initiales non nulles, la figure 5.5 montre que le système converge et qu'il y a plus d'oscillations de la vitesse au démarrage.

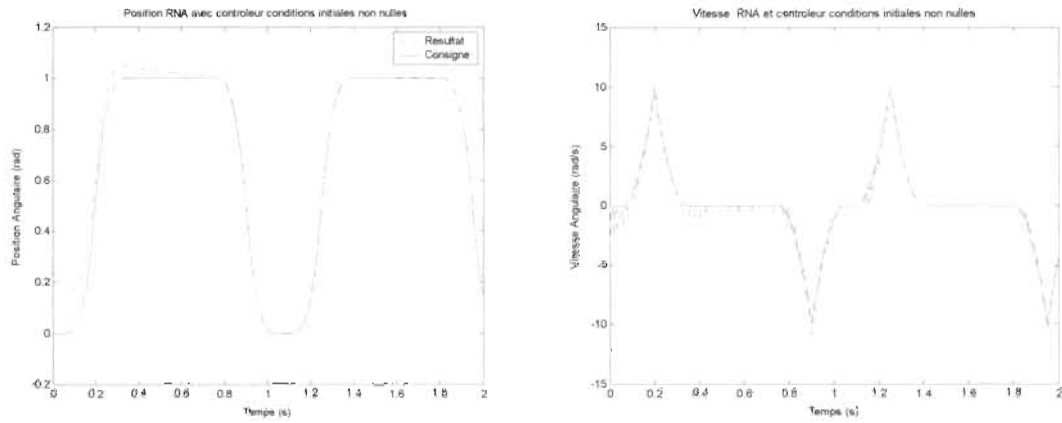


Figure 5.9 Résultats avec RNA et contrôleur MG et conditions initiales non nulles.

La figure 5.10 représente le couple de commande. Notons la grande valeur au début pour corriger la grande erreur de position qui semble induire plus d'oscillations. Notons aussi que le système est stable (les oscillations convergent vers 0).

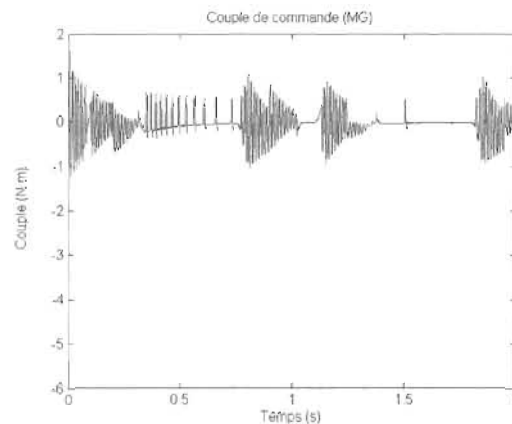


Figure 5.10 Couple de commande à la sortie du contrôleur MG et conditions initiales non nulles.

La figure 5.11 montre la convergence de l'erreur vers zéro à partir d'une grande valeur de l'erreur de position initiale

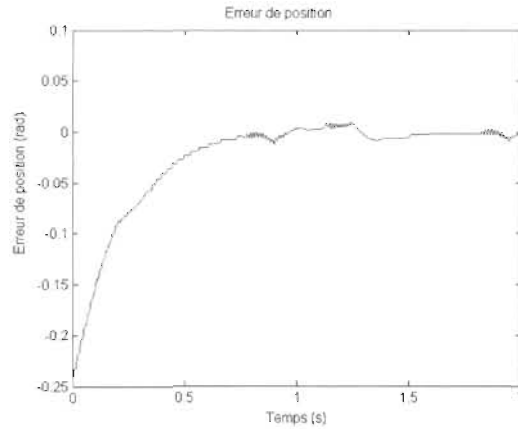


Figure 5.11 Erreur de position avec contrôleur MG et conditions initiales non nulles.

Les indices de performance pour le système avec RNA d'anticipation avec le contrôleur MG sont présentés dans le tableau 5.1. Notons l'amélioration tangible, avec les conditions initiales non nulles, où les indices sont 30 et 200 fois plus petits que sans le contrôleur. Dans le cas des conditions initiales non nulles, les valeurs des indices de performance restent grandes à cause de la grande valeur d'erreur de position au démarrage.

5.1.3 Réponse à la consigne avec contrôleur MG avec une rétroaction du terme de vitesse du moteur

Comme on a constaté dans le paragraphe précédent, le contrôleur induit des oscillations non voulues dans la trajectoire de vitesse. L'ajout d'un terme en vitesse du moteur dans la structure du contrôleur semble suffisant pour les minimiser. La valeur $K_v = 0.01$ est choisie pour la meilleure performance (Annexe C). La figure 5.12 montre les résultats de simulation. Notons la diminution des oscillations.

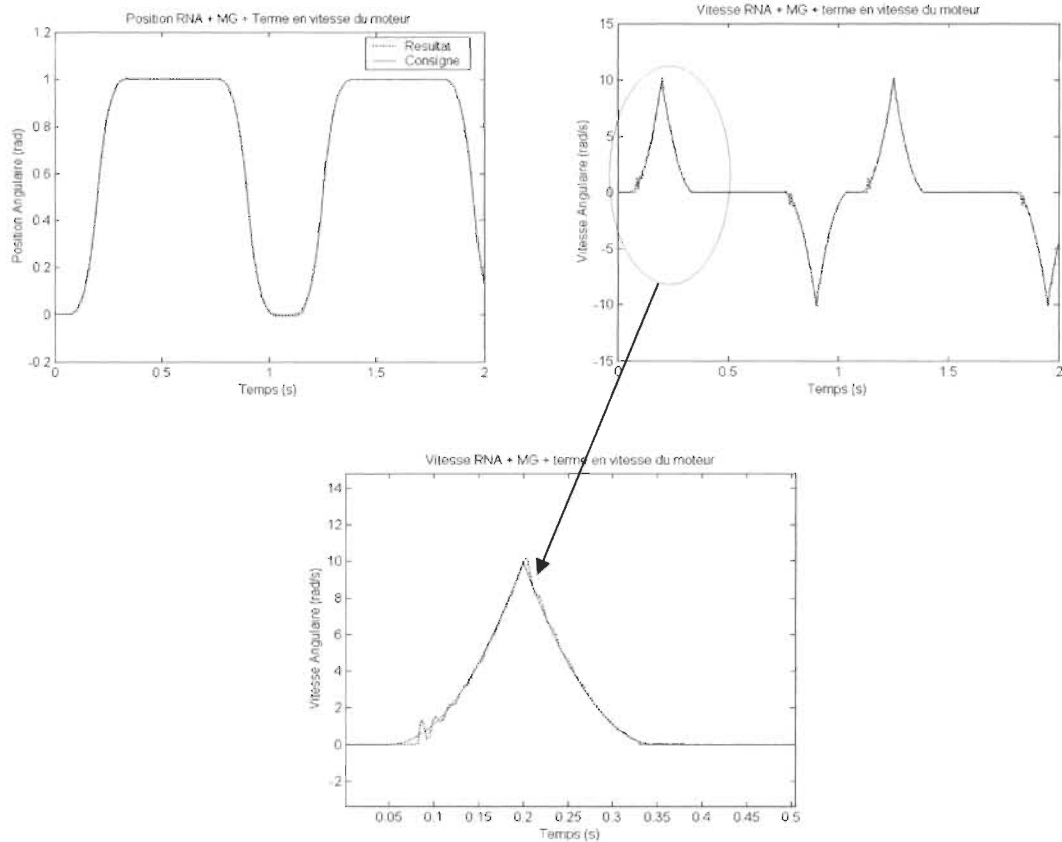


Figure 5.12 Résultats avec RNA et contrôleur MG incluant la rétroaction de vitesse du moteur.

La figure 5.13 montre le signal de commande. On constate que les oscillations ont diminué en amplitude et en durée.

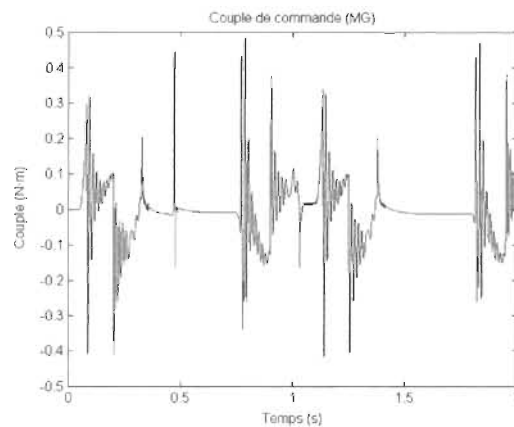


Figure 5.13 Couple de commande à la sortie du contrôleur MG incluant la rétroaction de vitesse du moteur.

La figure 5.14 montre l'erreur de position, notons qu'il y a moins d'oscillations mais la valeur d'erreur est plus élevée que sans la rétroaction de vitesse du moteur.

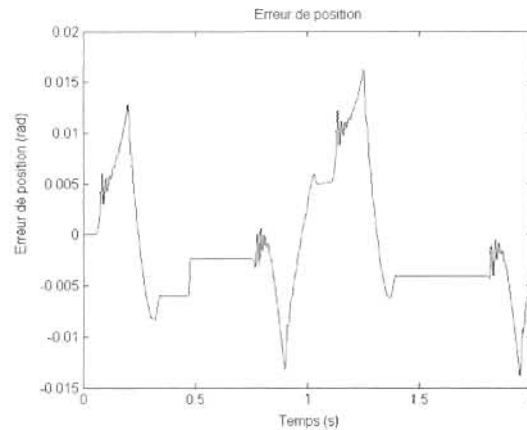


Figure 5.14 Erreur de position avec contrôleur MG incluant la rétroaction de vitesse du moteur.

La figure 5.15 montre les résultats avec conditions initiales non nulles. On constate la convergence.

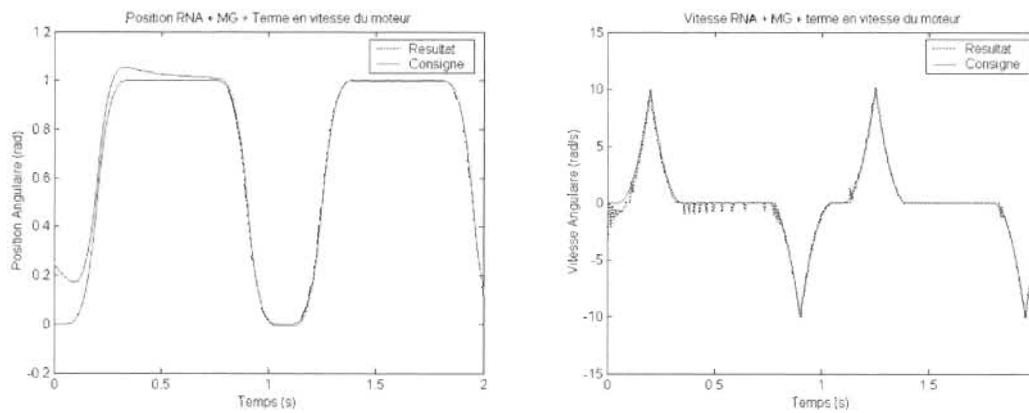


Figure 5.15 Résultats avec RNA et contrôleur MG incluant un terme en vitesse du moteur et conditions initiales non nulles.

La figure 5.16 représente le couple de commande où il y a une grande activité au début pour corriger la grande erreur initiale.

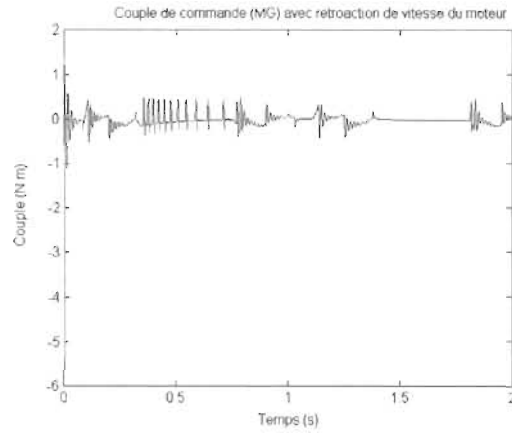


Figure 5.16 Couple de commande à la sortie du contrôleur MG incluant un terme en vitesse du moteur et conditions initiales non nulles.

La figure 5.17 montre la convergence de l'erreur vers zéro à partir d'une grande valeur de l'erreur de position initiale.

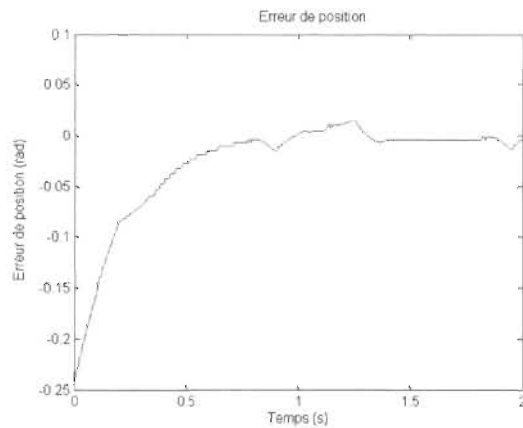


Figure 5.17 Erreur de position avec contrôleur MG incluant un terme en vitesse du moteur et conditions initiales non nulles.

Les indices de performance pour le système avec RNA d'anticipation avec le contrôleur MG incluant un terme en vitesse du moteur et dans des conditions initiales non nulles, sont présentés dans le tableau 5.1. Un léger compromis de la performance de suivi de la consigne de position a permis d'augmenter la stabilisation interne et de minimiser les

oscillations de vitesse. Dans le cas des conditions initiales non nulles, les valeurs des indices restent grandes à cause de la grande valeur d'erreur de position au démarrage.

Dans le tableau 5.1 se trouve le sommaire des résultats de simulation de l'algorithme de commande proposée.

Tableau 5.1 Indices de performance

<u>Conditions de simulation</u>	<u>Conditions initiales</u>	<u>Max e </u>	<u>$\sqrt{\sum e^2}$</u>
Réponse à la consigne sans le contrôleur	Conditions initiales nulles	0.214	10.0337
	Conditions initiales non nulles	0.25	9.7805
Réponse à la consigne avec contrôleur MG	Conditions initiales nulles	0.0099	0.3142
	Conditions initiales non nulles	0.25	7.9955
Réponse à la consigne avec contrôleur MG avec une rétroaction du terme de vitesse du moteur	Conditions initiales nulles	0.0149	0.4571
	Conditions initiales non nulles	0.25	7.9722

Pour vérifier la robustesse du contrôleur MG face aux variations des paramètres et de la charge, deux simulations sont réalisées avec conditions initiales nulles. La figure 5.18 montre les résultats de simulation. Dans a), on montre la performance avec l'inertie de la charge J_{ch} augmentée d'un facteur de 1.5. Les indices de performances sont $\mathbf{Max}|e| = 0.0205$ et $\sqrt{\sum e^2} = 0.9118$. Dans b), on montre la performance avec l'inertie de la charge J_{ch} et la friction de Coulomb augmentées toutes deux d'un facteur de 1.5. Les indices de performances sont $\mathbf{Max}|e| = 0.0186$ et $\sqrt{\sum e^2} = 0.8482$

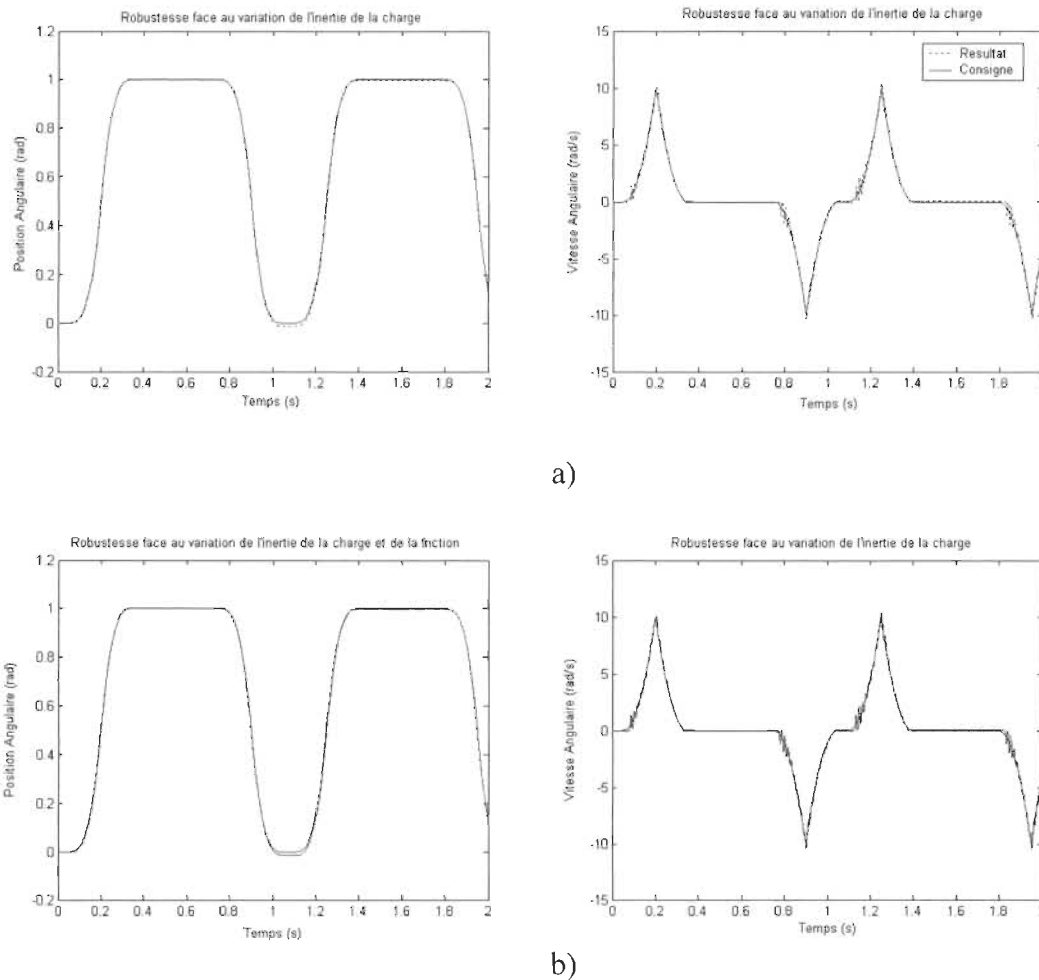


Figure 5.18 Robustesse du contrôleur face aux variations des paramètres et de la charge.

5.2 Conclusion

Dans ce chapitre on a présenté les résultats de simulation qui montrent une bonne performance et une poursuite assez précise de la consigne de position. Les limites de la loi de commande proposée sont aussi montrées comme par exemple : la grande phase de convergence quand les conditions initiales ne sont pas nulles, les oscillations dans la trajectoire de vitesse.

Une solution qui contient un compromis entre l'erreur de position et les oscillations dans le signal de commande est montrée, en introduisant la rétroaction de la vitesse du moteur dans la boucle de commande.

Chapitre 6

Conclusion

Dans ce mémoire nous avons présenté un algorithme de commande pour les articulations robotiques flexibles en présence de friction. Cet algorithme est basé sur l'utilisation d'un RNA auquel on a fait apprendre hors ligne le modèle inverse du joint flexible.

Les méthodes classiques d'identification du modèle inverse sont difficiles à utiliser à cause de la non linéarité dure autour de la vitesse zéro causée par le modèle de friction. Le RNA est introduit comme outil pour identifier le modèle inverse du système. Cet apprentissage est fait hors ligne avec une base de données (des entrées-sorties du système) qui sont générées par simulation en amenant l'articulation à réaliser la trajectoire désirée.

Les résultats de simulations avec le modèle inverse appris hors ligne sans boucle fermée de commande montrent de piètres performances. Alors, un contrôleur en mode de glissement (MG) est utilisé.

Pour minimiser les oscillations causées par le signal de commande MG, une rétroaction de la vitesse du moteur est ajoutée dans la boucle de commande mais avec le compromis d'une légère augmentation de l'erreur de position.

L'algorithme de commande présentée montre sa stabilité (les oscillations convergent vers 0) et sa robustesse quand les conditions initiales ne sont pas nulles.

L'apprentissage hors ligne présente des limites. Un travail futur peut porter sur l'utilisation des méthodes d'apprentissage en ligne pour tenir compte de la dynamique du système, des perturbations externes et du bruit.

Bibliographie

- [1] Sweet L.M. et Good M. C. "Redefinition of the robot motion-control problem" IEEE Control System Magazine, août 1985, 18-25.
- [2] Spong, M. W. " Control of Flexible Joint Robots: A Survey " No. de document: UILU-ENG-90-2203 DC-116 University of Illinois at Urbana-Champaign, février 1990.
- [3] Sicard, Pierre "Commande de position d'un moteur à courant continu par correcteurs adaptatifs développés à l'aide de la théorie du réglage par mode de glissement" Mémoire de maîtrise en électronique industrielle, UQTR, 1989.
- [4] Slotine J. E. "Tracking Control of nonlinear systems using sliding surfaces" Thèse de Doctorat, MIT 1981. non publiée, MIT, 1983.
- [5] Hached, M., Madani-Eseahani, S. M. et Zak, S. H. "Stabilization of Uncertain Systems Subject to Hard Bounds on Control with Application to a Manipulator" IEEE Journal of Robotics and Automation, 1988, Vol. 4, No. 3, 310-323.
- [6] Armstrong W.W. "Recursive Solution to Equations of Motion of an n-link manipulator " Proc. 5th World Congress: Theory of Machines and Mechanisms, Vol.102, pp. 1343-1346, Juillet 1979.
- [7] Wang, D. "Trajectory Tracking for Manipulators with Flexible Joints Using Link Variable Feedback" Proc. 32nd CDC, San Antonio TX, 633-637, Décembre 1993.
- [8] Qu, Z. "Input Output Robust Tracking Control Design for Flexible Joint Robots" IEEE Transactions on Automatic Control, AC-40(1), pp. 78-83, Janvier 1995.

- [9] Al-Ashoor, E.A., Patel R.V. et Khorasani K. "Robust Adaptive Controller Design and Stability Analysis for Flexible Joint Manipulators" IEEE Transactions on Systems, Man and Cybernetics, SMC-23(2), 589-602, Mars/Avril 1993.
- [10] Kuschewski J.G. et Zak S. H, "Application of Feedforward Neural Networks to Dynamical System Identification and Control" IEEE Transactions on Control Systems Technology, Vol.1, No. 1, mars 1993, 37-49.
- [11] Critchlow A.J., "Introduction to robotics" Macmillan Publishing company, New-York, 1985.
- [12] Sicard P. "Trajectory tracking of flexible joint manipulators with passivity" thèse doctorale, Rensselaer Polytechnic Institute, Juin 1993.
- [13] Sidi E. Y. O., Sicard P., Massicotte D. et Lesueur S., "Adaptive high precision position control for a flexible joint with friction and parameter uncertainties using neural networks" IEEE Congrès Canadien en Génie Électrique et Informatique, Mai 1998, Waterloo ON.
- [14] Canudas de Wit C. et al. "A new model for control of systems with friction" IEEE Transaction on Automatic Control, Vol. 40, No. 3, Mars 1995.
- [15] Armstrong B. et Canudas de Wit C. "Friction modeling", Control Handbook, W.S. Levine. éditeur, IEEE Press., 1995.
- [16] Lesueur S. "Compensation de friction par loi de commande adaptative indirecte" Rapport de cours de Théorie des systèmes asservis échantillonnés linéaires et non linéaires, Automne 1997.

- [17] Canudas de Wit C., Noël, P., Aubin, A., et Brogliato, B., "Adaptative friction compensation in robot manipulators : low velocities" Int. J. Robotics Res., 10(3), 189-99, 1991.
- [18] Vidal, Martin, "Architecture systolique pour un algorithme basé sur les réseaux de neurones pour l'égalisation de canaux" Mémoire de maîtrise. Université du Québec à Trois-Rivières, 1999.
- [19] Buhler, Hansuedi, "Réglage par logique floue" Lausanne : Presse polytechniques et universitaires romandes, 1994.
- [20] Chebassier, Joël, "Méthodologies pour la conception d'un système de commande par ordinateur " Thèse de Doctorat, Institut National Polytechnique de Grenoble, Janvier 1999.
- [21] Hirschorn R. M. "Invertibility of Nonlinear Control Systems" SIAM J. Control & Optimization, Vol.17, No.2, Mars 1979.
- [22] Stanislaw Hebertt J. Ramirez, Zak H. "The Adaptation Of Perceptrons with Applications to Inverse Dynamics Identification Of Unknown Dynamic Systems" IEEE Trans. Sys. Man Cyb., Vol. 21, No. 3, Mai/Juin 1991, pp. 634-64
- [23] Fukuda Toshio et Shibata Takanori "Theory and Application of Neural Networks for Industrial Control Systems" IEEE Trans. Industrial Electronic, Vol.39, No.6, Décembre 1992, pp. 472-489
- [24] "Neural Network Toolbox User's Manual" le guide d'utilisateur de Matlab®, MathWorks Inc.

Annexe A

Programmation du Réseau de Neurones

On montre ici le programme Matlab® utilisé pour créer le réseau de neurones et lui faire apprendre le modèle inverse de l'articulation flexible avec friction.

```
%          Le modèle inverse de l'articulation          %
%          %
%          Programme d'apprentissage          %

clear all ;
close all ;

%          Les données utilisées pour l'apprentissage  %
load anti ;

acht      = aTarget(1:55000) ;
wcht      = wTarget(1:55000);
thetacht  = Target(1:55000) ;
torquecht = Tm(1:55000) ;
pm        = [thetacht';wcht';acht'] ;
t         = torquecht' ;

%          La création du réseau          %
net = newfftd (minmax (pm), [0 2], [6 10 1], {'satlin', 'tansig', 'purelin'}, 'trainlm');
net.trainParam.goal = 1e-10;
net.trainParam.epochs = 100;
net.trainParam.show = 10;
net.trainParam.mem_reduc = 3;
net.performFcn = 'SSE';

achv      = aTarget(55001:60001);
wchv      = wTarget(55001:60001);
thetachv  = Target(55001:60001);
torquechv = Tm(55001:60001);
VV.P      = [thetachv';wchv';achv'];
VV.T      = torquechv';

%          L'apprentissage du réseau          %
net = train(net,pm,t,[ ],[ ],VV);
```

Annexe B

Approximation de la fonction « sign »

Logique Floue :

On a utilisé la logique floue pour l'adoucissement de la fonction Signe simplement comme un gain variable presque linéaire en utilisant la logiciel Matlab®

```
1. If (s is nnnn) then (Sign(s) is nnn) (1)
2. If (s is nnn) then (Sign(s) is nnn) (1)
3. If (s is nn) then (Sign(s) is nn) (1)
4. If (s is n) then (Sign(s) is n) (1)
5. If (s is z) then (Sign(s) is z) (1)
6. If (s is p) then (Sign(s) is p) (1)
7. If (s is pp) then (Sign(s) is pp) (1)
8. If (s is ppp) then (Sign(s) is ppp) (1)
9. If (s is pppp) then (Sign(s) is ppp) (1)
```

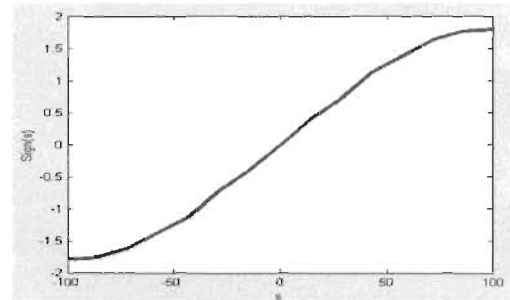


Figure A.1 Approximation floue de la fonction Signe sur Matlab®

Annexe C

Syntonisation des gains du contrôleur MG

Ce qui suit c'est le programme Matlab® utilisé pour la sélection des gains pour le contrôleur par MG ainsi que les résultats obtenus de ce programme.

```
%
%

clear all;
load nanti;
Jch = 5.5e-5;
KA=[0.3 0.4 0.5 0.6 0.7];
K0A=[18 19 20 21 22];
K1A=[2.8 3 3.5 3.8 4.2];
KvA=[0 0.001 0.01 0.05 0.1];

for i=1:5
    for j=1:5
        for n=1:5
            for m=1:5
                K=KA(m);
                K0=K0A(n);
                K1=K1A(j);
                Kv=KvA(i);

sim('testffnew')% sim('testanti')
delta=thetad-thetach;
delta2=realpow(delta,2);
sumdelta2=sum(delta2);
qrtsumdelta2(i,j,n,m)=sqrt(sumdelta2);
maxabserreur(i,j,n,m)=max(abs(delta));
            end
        end
    end
end
```


Les valeurs absolues maximales des erreurs de position de la charge ($\text{Max}|e|$)

		K = 0.3				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 18	Kv = 0	0.0175	0.0176	0.0178	0.0179	0.0181
	Kv = 0.001	0.0183	0.0184	0.0186	0.0186	0.0187
	Kv = 0.01	0.0241	0.0241	0.0242	0.0243	0.0244
	Kv = 0.05	0.0467	0.0468	0.0468	0.0469	0.047
	Kv = 0.1	0.0706	0.0706	0.0707	0.0708	0.0709

		K = 0.3				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 19	Kv = 0	0.0172	0.0173	0.0176	0.0177	0.0179
	Kv = 0.001	0.018	0.0181	0.0183	0.0184	0.0185
	Kv = 0.01	0.0238	0.0238	0.0239	0.024	0.0241
	Kv = 0.05	0.046	0.0461	0.0462	0.0463	0.0463
	Kv = 0.1	0.0696	0.0697	0.0698	0.0698	0.0699

		K = 0.3				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 20	Kv = 0	0.017	0.0171	0.0174	0.0175	0.0177
	Kv = 0.001	0.0178	0.0179	0.0181	0.0182	0.0183
	Kv = 0.01	0.0235	0.0235	0.0237	0.0237	0.0238
	Kv = 0.05	0.0454	0.0454	0.0456	0.0456	0.0457
	Kv = 0.1	0.0687	0.0687	0.0688	0.0689	0.069

		K = 0.3				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 21	Kv = 0	0.0168	0.0169	0.0171	0.0173	0.0175
	Kv = 0.001	0.0175	0.0176	0.0179	0.018	0.0182
	Kv = 0.01	0.0232	0.0233	0.0234	0.0235	0.0235
	Kv = 0.05	0.0448	0.0448	0.0449	0.045	0.0451
	Kv = 0.1	0.0678	0.0678	0.0679	0.068	0.068

		K = 0.3				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 22	Kv = 0	0.0166	0.0167	0.0169	0.0171	0.0173
	Kv = 0.001	0.0173	0.0174	0.0177	0.0178	0.018
	Kv = 0.01	0.023	0.023	0.0231	0.0232	0.0233
	Kv = 0.05	0.0441	0.0442	0.0443	0.0444	0.0445
	Kv = 0.1	0.0669	0.0669	0.067	0.0671	0.0672

		K = 0.4				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1= 3.8	K1 = 4
K0 = 18	Kv = 0	0.0148	0.0148	0.0149	0.015	0.0151
	Kv = 0.001	0.0152	0.0153	0.0154	0.0155	0.0156
	Kv = 0.01	0.0202	0.0202	0.0203	0.0204	0.0205
	Kv = 0.05	0.0393	0.0393	0.0394	0.0395	0.0396
	Kv = 0.1	0.0606	0.0606	0.0607	0.0608	0.0608

		K = 0.4				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1= 3.8	K1 = 4
K0 = 19	Kv = 0	0.0146	0.0147	0.0148	0.0149	0.015
	Kv = 0.001	0.0151	0.0152	0.0153	0.0154	0.0155
	Kv = 0.01	0.02	0.02	0.0201	0.0202	0.0203
	Kv = 0.05	0.0389	0.0389	0.039	0.039	0.0391
	Kv = 0.1	0.0599	0.0599	0.06	0.0601	0.0601

		K = 0.4				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1= 3.8	K1 = 4
K0 = 20	Kv = 0	0.0145	0.0146	0.0147	0.0147	0.0148
	Kv = 0.001	0.015	0.015	0.0152	0.0153	0.0154
	Kv = 0.01	0.0198	0.0199	0.02	0.02	0.0201
	Kv = 0.05	0.0384	0.0385	0.0385	0.0386	0.0386
	Kv = 0.1	0.0592	0.0592	0.0593	0.0594	0.0594

		K = 0.4				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1= 3.8	K1 = 4
K0 = 21	Kv = 0	0.0143	0.0144	0.0145	0.0146	0.0147
	Kv = 0.001	0.0149	0.0149	0.015	0.0151	0.0152
	Kv = 0.01	0.0197	0.0197	0.0198	0.0198	0.0199
	Kv = 0.05	0.038	0.038	0.0381	0.0382	0.0382
	Kv = 0.1	0.0585	0.0585	0.0586	0.0587	0.0588

		K = 0.4				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1= 3.8	K1 = 4
K0 = 22	Kv = 0	0.014	0.0141	0.0143	0.0144	0.0145
	Kv = 0.001	0.0147	0.0148	0.0149	0.015	0.0151
	Kv = 0.01	0.0195	0.0195	0.0196	0.0197	0.0198
	Kv = 0.05	0.0376	0.0376	0.0377	0.0378	0.0378
	Kv = 0.1	0.0579	0.0579	0.058	0.058	0.0581

		K = 0.5				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1= 3.8	K1 = 4
K0 =18	Kv = 0	0.0115	0.0117	0.0121	0.0123	0.0125
	Kv = 0.001	0.0122	0.0124	0.0127	0.0129	0.0131
	Kv = 0.01	0.0174	0.0174	0.0175	0.0176	0.0176
	Kv = 0.05	0.0342	0.0342	0.0343	0.0343	0.0344
	Kv = 0.1	0.0531	0.0531	0.0532	0.0532	0.0533

		K = 0.5				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1= 3.8	K1 = 4
K0 =19	Kv = 0	0.0116	0.0117	0.0121	0.0122	0.0125
	Kv = 0.001	0.012	0.0122	0.0125	0.0127	0.013
	Kv = 0.01	0.0172	0.0173	0.0174	0.0175	0.0175
	Kv = 0.05	0.0339	0.0339	0.034	0.034	0.0341
	Kv = 0.1	0.0525	0.0526	0.0526	0.0527	0.0527

		K = 0.5				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1= 3.8	K1 = 4
K0 =20	Kv = 0	0.0118	0.0118	0.0121	0.0122	0.0124
	Kv = 0.001	0.0119	0.0121	0.0124	0.0126	0.0129
	Kv = 0.01	0.0171	0.0171	0.0173	0.0173	0.0174
	Kv = 0.05	0.0335	0.0336	0.0336	0.0337	0.0337
	Kv = 0.1	0.052	0.052	0.0521	0.0522	0.0522

		K = 0.5				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1= 3.8	K1 = 4
K0 =21	Kv = 0	0.0118	0.0119	0.0121	0.0123	0.0124
	Kv = 0.001	0.0118	0.0121	0.0124	0.0126	0.0128
	Kv = 0.01	0.0169	0.017	0.0171	0.0172	0.0173
	Kv = 0.05	0.0332	0.0333	0.0333	0.0334	0.0334
	Kv = 0.1	0.0515	0.0515	0.0516	0.0516	0.0517

		K = 0.5				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1= 3.8	K1 = 4
K0 =22	Kv = 0	0.0117	0.012	0.0122	0.0123	0.0124
	Kv = 0.001	0.0118	0.012	0.0123	0.0125	0.0127
	Kv = 0.01	0.0168	0.0169	0.017	0.0171	0.0171
	Kv = 0.05	0.0329	0.033	0.033	0.0331	0.0331
	Kv = 0.1	0.051	0.051	0.0511	0.0511	0.0512

		K = 0.6				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 18	Kv = 0	0.0104	0.0104	0.0108	0.011	0.0113
	Kv = 0.001	0.0111	0.0112	0.0115	0.0117	0.0119
	Kv = 0.01	0.0151	0.0151	0.0152	0.0153	0.0153
	Kv = 0.05	0.0302	0.0303	0.0303	0.0304	0.0304
	Kv = 0.1	0.0473	0.0473	0.0474	0.0474	0.0474

		K = 0.6				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 19	Kv = 0	0.0103	0.0102	0.0106	0.0109	0.0111
	Kv = 0.001	0.0109	0.011	0.0113	0.0115	0.0117
	Kv = 0.01	0.015	0.015	0.0151	0.0152	0.0152
	Kv = 0.05	0.03	0.03	0.0301	0.0301	0.0302
	Kv = 0.1	0.0468	0.0469	0.0469	0.047	0.047

		K = 0.6				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 20	Kv = 0	0.0112	0.0101	0.0104	0.0106	0.0109
	Kv = 0.001	0.0106	0.0108	0.0111	0.0113	0.0115
	Kv = 0.01	0.0149	0.0149	0.015	0.0151	0.0152
	Kv = 0.05	0.0298	0.0298	0.0299	0.0299	0.03
	Kv = 0.1	0.0464	0.0465	0.0465	0.0466	0.0466

		K = 0.6				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 21	Kv = 0	0.0118	0.0101	0.0101	0.0104	0.0107
	Kv = 0.001	0.0104	0.0106	0.0109	0.0111	0.0113
	Kv = 0.01	0.0148	0.0148	0.0149	0.015	0.0151
	Kv = 0.05	0.0296	0.0296	0.0296	0.0297	0.0297
	Kv = 0.1	0.046	0.0461	0.0461	0.0462	0.0462

		K = 0.6				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 22	Kv = 0	0.0129	0.0111	0.0099	0.0102	0.0105
	Kv = 0.001	0.0102	0.0103	0.0107	0.0109	0.0112
	Kv = 0.01	0.0147	0.0148	0.0149	0.0149	0.015
	Kv = 0.05	0.0293	0.0294	0.0294	0.0295	0.0295
	Kv = 0.1	0.0456	0.0457	0.0457	0.0458	0.0458

		K = 0.7				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 18	Kv = 0	0.0859	0.0145	0.0114	0.0102	0.0101
	Kv = 0.001	0.0127	0.0124	0.0095	0.0099	0.0103
	Kv = 0.01	0.0135	0.0135	0.0136	0.0136	0.0137
	Kv = 0.05	0.0271	0.0272	0.0272	0.0273	0.0273
	Kv = 0.1	0.0424	0.0424	0.0425	0.0425	0.0426

		K = 0.7				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 19	Kv = 0	0.088	0.0841	0.0125	0.0104	0.0101
	Kv = 0.001	0.0129	0.0126	0.0098	0.0099	0.0103
	Kv = 0.01	0.0134	0.0135	0.0135	0.0136	0.0136
	Kv = 0.05	0.027	0.027	0.027	0.0271	0.0271
	Kv = 0.1	0.0421	0.0421	0.0421	0.0422	0.0423

		K = 0.7				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 20	Kv = 0	0.0881	0.0848	0.0128	0.0106	0.0102
	Kv = 0.001	0.0137	0.0126	0.0101	0.01	0.0103
	Kv = 0.01	0.0134	0.0134	0.0135	0.0135	0.0136
	Kv = 0.05	0.0268	0.0268	0.0269	0.0269	0.0269
	Kv = 0.1	0.0418	0.0418	0.0418	0.0419	0.0419

		K = 0.7				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 21	Kv = 0	0.0882	0.0866	0.0128	0.0108	0.0102
	Kv = 0.001	0.015	0.0129	0.0103	0.0101	0.0104
	Kv = 0.01	0.0133	0.0133	0.0134	0.0135	0.0135
	Kv = 0.05	0.0266	0.0266	0.0267	0.0267	0.0268
	Kv = 0.1	0.0414	0.0415	0.0415	0.0416	0.0416

		K = 0.7				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 22	Kv = 0	0.0904	0.0876	0.0132	0.0119	0.0102
	Kv = 0.001	0.0864	0.0134	0.0104	0.0102	0.0103
	Kv = 0.01	0.0132	0.0133	0.0134	0.0134	0.0135
	Kv = 0.05	0.0264	0.0265	0.0265	0.0265	0.0266
	Kv = 0.1	0.0411	0.0412	0.0412	0.0412	0.0413

Les valeurs absolues maximaux des erreurs de position du charge ($\sqrt{\sum e^2}$)

		K = 0.3				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 18	Kv = 0	0.5391	0.5434	0.5536	0.5595	0.5667
	Kv = 0.001	0.5649	0.5687	0.577	0.5805	0.5864
	Kv = 0.01	0.7621	0.7638	0.7682	0.771	0.7752
	Kv = 0.05	1.6394	1.6407	1.6408	1.6421	1.6432
	Kv = 0.1	2.6113	2.6126	2.6154	2.6164	2.6179

		K = 0.3				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 19	Kv = 0	0.5317	0.5362	0.5466	0.5527	0.5604
	Kv = 0.001	0.5565	0.5615	0.571	0.5762	0.5808
	Kv = 0.01	0.7539	0.7552	0.7602	0.763	0.7667
	Kv = 0.05	1.6157	1.6189	1.6205	1.6223	1.625
	Kv = 0.1	2.5754	2.5785	2.5809	2.5834	2.5863

		K = 0.3				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 20	Kv = 0	0.5239	0.5275	0.5398	0.5461	0.5539
	Kv = 0.001	0.5502	0.5545	0.5643	0.5703	0.5755
	Kv = 0.01	0.7461	0.7477	0.7522	0.7488	0.7584
	Kv = 0.05	1.5972	1.5984	1.6019	1.6035	1.6057
	Kv = 0.1	2.5442	2.5423	2.5457	2.5506	2.5503

		K = 0.3				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 21	Kv = 0	0.5178	0.5221	0.5326	0.5393	0.5478
	Kv = 0.001	0.5428	0.5468	0.5576	0.5636	0.5725
	Kv = 0.01	0.7381	0.7401	0.7442	0.7471	0.7505
	Kv = 0.05	1.5769	1.5764	1.5796	1.5816	1.5843
	Kv = 0.1	2.5112	2.5135	2.5157	2.5154	2.5203

		K = 0.3				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 22	Kv = 0	0.5135	0.5168	0.5254	0.5331	0.5427
	Kv = 0.001	0.5348	0.5401	0.551	0.5579	0.5669
	Kv = 0.01	0.7311	0.733	0.7371	0.7396	0.743
	Kv = 0.05	1.5576	1.5589	1.5623	1.5642	1.5664
	Kv = 0.1	2.478	2.4816	2.4824	2.4843	2.4893

		K = 0.4				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 18	Kv = 0	0.4461	0.4454	0.4519	0.4558	0.4605
	Kv = 0.001	0.4602	0.4629	0.4692	0.4731	0.4772
	Kv = 0.01	0.6351	0.6366	0.64	0.6421	0.645
	Kv = 0.05	1.3743	1.3752	1.3757	1.3777	1.3814
	Kv = 0.1	2.2215	2.2228	2.2234	2.2278	2.2301

		K = 0.4				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 19	Kv = 0	0.4387	0.4421	0.4478	0.4515	0.4563
	Kv = 0.001	0.4561	0.4587	0.465	0.4688	0.4736
	Kv = 0.01	0.6295	0.6313	0.6349	0.6369	0.6337
	Kv = 0.05	1.3604	1.3615	1.3636	1.365	1.3653
	Kv = 0.1	2.1945	2.1981	2.2015	2.2011	2.2051

		K = 0.4				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 20	Kv = 0	0.4337	0.4375	0.4439	0.4476	0.4524
	Kv = 0.001	0.4521	0.4547	0.4609	0.4646	0.4695
	Kv = 0.01	0.6244	0.6203	0.6298	0.6318	0.6345
	Kv = 0.05	1.3448	1.3478	1.35	1.3498	1.3529
	Kv = 0.1	2.1725	2.1743	2.1752	2.1792	2.1795

		K = 0.4				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 21	Kv = 0	0.425	0.4324	0.4405	0.4435	0.4487
	Kv = 0.001	0.4482	0.4509	0.4571	0.4606	0.4642
	Kv = 0.01	0.6211	0.6209	0.6248	0.6268	0.6295
	Kv = 0.05	1.3317	1.3343	1.3364	1.338	1.3401
	Kv = 0.1	2.1494	2.1483	2.1539	2.1562	2.1587

		K = 0.4				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 22	Kv = 0	0.4198	0.4245	0.4372	0.4398	0.4449
	Kv = 0.001	0.445	0.4471	0.4533	0.4569	0.4615
	Kv = 0.01	0.6168	0.6183	0.62	0.6219	0.6244
	Kv = 0.05	1.3202	1.3215	1.322	1.3255	1.3271
	Kv = 0.1	2.1244	2.1281	2.129	2.1312	2.1355

		K = 0.5				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 18	Kv = 0	0.3748	0.3744	0.3809	0.3882	0.3945
	Kv = 0.001	0.3912	0.393	0.4	0.4044	0.411
	Kv = 0.01	0.5423	0.5447	0.5486	0.5505	0.5472
	Kv = 0.05	1.1856	1.1849	1.1886	1.1885	1.1922
	Kv = 0.1	1.923	1.9242	1.9263	1.9271	1.9293

		K = 0.5				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 19	Kv = 0	0.3684	0.3852	0.3799	0.383	0.3896
	Kv = 0.001	0.3829	0.3905	0.3938	0.4005	0.4067
	Kv = 0.01	0.5385	0.5405	0.5445	0.5468	0.544
	Kv = 0.05	1.1741	1.1765	1.1772	1.1802	1.1816
	Kv = 0.1	1.9054	1.906	1.9087	1.91	1.9113

		K = 0.5				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 20	Kv = 0	0.3536	0.3626	0.3787	0.3788	0.3862
	Kv = 0.001	0.3799	0.3825	0.393	0.3973	0.4033
	Kv = 0.01	0.5342	0.5358	0.5408	0.543	0.5401
	Kv = 0.05	1.1663	1.1671	1.1688	1.1703	1.1705
	Kv = 0.1	1.886	1.8892	1.8915	1.8912	1.8945

		K = 0.5				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 21	Kv = 0	0.3593	0.3549	0.3767	0.3796	0.3853
	Kv = 0.001	0.3775	0.3819	0.39	0.3939	0.3968
	Kv = 0.01	0.5302	0.5322	0.5366	0.5335	0.542
	Kv = 0.05	1.1494	1.1559	1.1599	1.1612	1.1629
	Kv = 0.1	1.8692	1.871	1.8742	1.876	1.8752

		K = 0.5				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 22	Kv = 0	0.3423	0.3582	0.3623	0.3782	0.3812
	Kv = 0.001	0.3891	0.379	0.3881	0.3908	0.3958
	Kv = 0.01	0.5258	0.5281	0.5326	0.5353	0.5385
	Kv = 0.05	1.1476	1.1486	1.1503	1.1502	1.1536
	Kv = 0.1	1.8527	1.8561	1.8557	1.8591	1.8612

		K = 0.6				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1= 3.8	K1 = 4
K0 =18	Kv = 0	0.3046	0.3085	0.3202	0.3301	0.3361
	Kv = 0.001	0.3189	0.3258	0.3438	0.3481	0.352
	Kv = 0.01	0.4702	0.4709	0.4729	0.4738	0.4758
	Kv = 0.05	1.0424	1.0424	1.0434	1.0444	1.0478
	Kv = 0.1	1.6997	1.7003	1.7026	1.7028	1.705

		K = 0.6				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1= 3.8	K1 = 4
K0 =19	Kv = 0	0.3035	0.3073	0.3143	0.3245	0.3346
	Kv = 0.001	0.3183	0.3236	0.338	0.343	0.3474
	Kv = 0.01	0.4675	0.4681	0.4704	0.4716	0.4731
	Kv = 0.05	1.0351	1.034	1.0378	1.0392	1.0389
	Kv = 0.1	1.6842	1.6871	1.6878	1.6904	1.6897

		K = 0.6				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1= 3.8	K1 = 4
K0 =20	Kv = 0	0.3128	0.3164	0.3143	0.3221	0.3298
	Kv = 0.001	0.3136	0.3222	0.338	0.3342	0.3418
	Kv = 0.01	0.4649	0.4659	0.4678	0.4691	0.4707
	Kv = 0.05	1.0281	1.0286	1.029	1.031	1.0336
	Kv = 0.1	1.6709	1.674	1.6758	1.6753	1.6786

		K = 0.6				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1= 3.8	K1 = 4
K0 =21	Kv = 0	0.3134	0.3093	0.3101	0.3168	0.3262
	Kv = 0.001	0.3144	0.3227	0.3248	0.3301	0.3434
	Kv = 0.01	0.4618	0.4572	0.4596	0.4664	0.4682
	Kv = 0.05	1.0133	1.022	1.024	1.0252	1.0262
	Kv = 0.1	1.6603	1.6607	1.661	1.6642	1.6658

		K = 0.6				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1= 3.8	K1 = 4
K0 =22	Kv = 0	0.3438	0.3082	0.3142	0.3213	0.3284
	Kv = 0.001	0.3238	0.3084	0.3256	0.3319	0.3402
	Kv = 0.01	0.4597	0.4547	0.4571	0.464	0.4657
	Kv = 0.05	1.0141	1.0155	1.0171	1.0181	1.0198
	Kv = 0.1	1.6473	1.6481	1.6506	1.6519	1.6513

		K = 0.7				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 18	Kv = 0	3.6012	0.4124	0.3115	0.3022	0.3178
	Kv = 0.001	0.35	0.3236	0.298	0.2998	0.3055
	Kv = 0.01	0.4168	0.4133	0.4214	0.4217	0.4231
	Kv = 0.05	0.9279	0.9292	0.9316	0.933	0.934
	Kv = 0.1	1.519	1.5178	1.522	1.5213	1.5229

		K = 0.7				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 19	Kv = 0	3.6908	3.3644	0.3189	0.32	0.3176
	Kv = 0.001	0.3754	0.3259	0.3029	0.302	0.3053
	Kv = 0.01	0.4142	0.4149	0.4178	0.4195	0.4209
	Kv = 0.05	0.9243	0.925	0.9267	0.9277	0.9292
	Kv = 0.1	1.5083	1.51	1.5094	1.5124	1.5126

		K = 0.7				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 20	Kv = 0	3.767	3.5794	0.3433	0.3315	0.317
	Kv = 0.001	0.3871	0.3522	0.2999	0.3176	0.3127
	Kv = 0.01	0.4115	0.4069	0.4143	0.4175	0.418
	Kv = 0.05	0.9192	0.9199	0.9214	0.9209	0.9238
	Kv = 0.1	1.4988	1.4999	1.5013	1.5018	1.5041

		K = 0.7				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 21	Kv = 0	3.8314	3.6421	0.3552	0.3203	0.3145
	Kv = 0.001	0.4087	0.3606	0.3048	0.3085	0.3139
	Kv = 0.01	0.4086	0.4096	0.4118	0.4151	0.4164
	Kv = 0.05	0.9136	0.9147	0.9163	0.9169	0.9186
	Kv = 0.1	1.489	1.49	1.4896	1.4907	1.494

		K = 0.7				
		K1 = 2.8	K1 = 3	K1 = 3.5	K1 = 3.8	K1 = 4
K0 = 22	Kv = 0	3.9148	3.6976	0.3742	0.3531	0.3241
	Kv = 0.001	3.4725	0.3763	0.322	0.3217	0.3164
	Kv = 0.01	0.4052	0.407	0.4099	0.4129	0.4145
	Kv = 0.05	0.9091	0.9097	0.9113	0.9121	0.9121
	Kv = 0.1	1.4793	1.4801	1.4798	1.4825	1.4841